

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

DEPARTMENT OF BIOMEDICAL ENGINEERING

**REGISTRACE ULTRAZVUKOVÝCH SEKVENCÍ S
VYUŽITÍM EVOLUČNÍCH ALGORITMŮ**

IMAGE REGISTRATION OF ULTRASOUND SEQUENCES USING EVOLUTIONARY ALGORITHMS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Bohdana Hnízdllová

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Mézl, Ph.D.

BRNO 2021

Diplomová práce

magisterský navazující studijní program **Biomedicínské inženýrství a bioinformatika**

Ústav biomedicínského inženýrství

Studentka: Bc. Bohdana Hnízdilová

ID: 195737

Ročník: 2

Akademický rok: 2020/21

NÁZEV TÉMATU:

Registrace ultrazvukových sekvencí s využitím evolučních algoritmů

POKYNY PRO VYPRACOVÁNÍ:

1) Seznamte se s evolučními algoritmy pro matematickou optimalizaci. Provedte rešerši využití těchto metod pro registraci obrazu. 2) Seznamte se s principem perfúzního zobrazování pomocí ultrazvuku a s problematikou registrace obrazu u ultrazvukových dat. 3) Implementujte vybranou kritériální funkci a alespoň jednu optimalizační metodu v programovém prostředí Matlab. Metody ověřte na testovacích sekvencích. 4) Práci rozšiřte o implementaci alespoň dvou evolučních algoritmů. 5) Navržené algoritmy otestujte na dynamických ultrazvukových sekvencích, které budou poskytnuty vedoucím práce. 6) Provedte vyhodnocení úspěšnosti registrace pomocí jednotlivých metod ve srovnání s manuální korekcí pohybu.

DOPORUČENÁ LITERATURA:

[1] DAMAS, S., O. CORDÓN a J. SANTAMARÍA. Medical Image Registration Using Evolutionary Computation: An Experimental Survey. IEEE Computational Intelligence Magazine. 2011, 6(4): 26-42. ISSN: 1556-603X.

[2] HAUPT, Randy L a Witold PEDRYCZ. Practical genetic algorithms: a comprehensive framework of computational intelligence. 2nd ed. Hoboken, N.J.: Wiley-Interscience, 2004, xvii, 253 s. ISBN 978-0-471-45-65-3.

Termín zadání: 8.2.2021

Termín odevzdání: 21.5.2021

Vedoucí práce: Ing. Martin Mézl, Ph.D.

prof. Ing. Ivo Provazník, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato diplomová práce se věnuje registraci ultrazvukových sekvencí s využitím evolučních algoritmů. V teoretické části práce je popsán proces registrace obrazů a jeho optimalizace pomocí genetických a metaheuristických algoritmů. V práci jsou také uvedeny problémy, které se mohou vyskytnout při registraci ultrasonografických snímků a různé přístupy k jejich registraci. V praktické části práce byla implementováno a srovnáno několik metod optimalizace pro registraci řady sekvencí.

KLÍČOVÁ SLOVA

registrace obrazů, ultrazvukové obrazy, optimalizační algoritmy, metaheuristika, genetické algoritmy, kritériální funkce, algoritmus úplného prohledávání, algoritmus roje částic, včelí algoritmus, algoritmus roje vážek, Nelder-Mead algoritmus

ABSTRACT

This master's thesis deals with the registration of ultrasound sequences using evolutionary algorithms. The theoretical part of the thesis describes the process of image registration and its optimization using genetic and metaheuristic algorithms. The thesis also presents problems that may occur during the registration of ultrasonographic images and various approaches to their registration. In the practical part of the work, several optimization methods for the registration of a number of sequences were implemented and compared.

KEYWORDS

image registration, ultrasound images, optimization algorithms, metaheuristics, genetic algorithms, objective function, exhaustive search, particle swarm optimization, bee algorithm, dragon fly algorithm, Nelder-Mead algorithm

HNÍZDILOVÁ, Bohdana. *Registrace ultrazvukových sekvencí s využitím evolučních algoritmů*. Brno, 2020, 85 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Martin Mézl, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Registrace ultrazvukových sekvencí s využitím evolučních algoritmů“ jsem vypracovala samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědoma následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autorky

PODĚKOVÁNÍ

Ráda bych poděkovala vedoucímu diplomové práce panu Ing. Martinu Mězlovi, Ph.D. za trpělivost, odborné vedení, konzultace, cenné rady a podnětné návrhy k práci.

Obsah

Úvod	11
1 Registrace obrazu	12
1.1 Postup při registraci obrazu	12
1.2 Povaha obrazu	13
1.2.1 Voxel-based (intensity-based)	13
1.2.2 Feature-based	14
1.3 Transformace scény	14
1.3.1 Rigidní transformace	15
1.3.2 Elastická transformace	16
1.3.3 Lokální transformace	17
1.4 Strategie prohledávání optimalizačního procesu	18
1.4.1 Matching-based RO	18
1.4.2 Parameters-based RO	18
1.5 Metrika podobnosti	19
2 Metody optimalizace	21
2.1 Metaheuristické a evoluční algoritmy	21
2.1.1 Simulované žíhání	21
2.1.2 Zakázané prohledávání	22
2.1.3 Nelder-Meadův algoritmus	23
2.1.4 Genetický algoritmus	24
2.1.5 Particle Swarm Optimalizace (PSO)	25
2.1.6 Včelí algoritmus (BA)	26
2.1.7 Dragonfly algoritmus (DA)	27
2.2 Gradientní metody	30
2.2.1 Metoda nejstrmějšího sestupu	30
2.3 Metody vyšších řádů	31
2.3.1 Newtonova metoda	31
3 Registrace obrazu u ultrazvukových dat	32
3.1 Odstraňování šumu	32
3.2 Zvýšení přesnosti registrace UZ obrazů	33
3.3 Perfúzní zobrazování	34
4 Realizace na simulovaných datech	36
4.1 Testovací sekvence	36
4.1.1 Předzpracování dat	36

4.1.2	Kriteriální funkce	38
4.1.3	Průběh kontrastu v sekvenci snímků	38
4.2	Metody optimalizace	39
4.2.1	Optimalizace metodou úplného prohledávání	39
4.2.2	Optimalizace metodou PSO	40
4.2.3	Optimalizace rojem včel (BA)	45
4.2.4	Optimalizace algoritmem vážek (DA)	49
4.3	Vyhodnocení a srovnání algoritmů na testovací sekvenci	53
5	Preklinická sekvence	57
5.1	Průběh kontrastu v sekvenci	57
5.2	Vyhodnocení procesu optimalizace a registrace	58
5.2.1	Výstup registrace	62
6	Klinická sekvence	63
6.1	Vyhodnocení procesu optimalizace a registrace	63
7	Klinická sekvence 2	66
7.1	Vyhodnocení procesu optimalizace a registrace	66
7.1.1	Výstup registrace	69
	Závěr	70
	Literatura	72
	Seznam příloh	77
A	Vyhodnocení optimalizace na simulovaných datech	78
A.1	Algoritmus PSO	78
A.2	Algoritmus BA	80
A.3	Algoritmus DA	82
B	Obsah odevzdaných příloh	85

Seznam obrázků

1.1	Optimalizační proces registrace obrazu, přeloženo z [1]	12
2.1	Představa Simplexu, převzato z [43]	23
2.2	Vývojový diagram genetického algoritmu	25
2.3	Pseudokód PSO algoritmu	26
2.4	Pseudokód BA algoritmu [40]	27
2.5	Vzory chování mezi vážkami v roji, upraveno z [35]	28
2.6	Pseudokód DA algoritmu [35]	30
4.1	Ukázka referenčního obrazu (vlevo) a transformovaného obrazu (vpravo)	37
4.2	Referenční snímek s obdélníkovým výřezem oblasti zájmu	37
4.3	Graf znázorňující měnící se průměrnou hodnotu jasu v sekvenci snímků, v závislosti na množství kontrastní látky	38
4.4	Grafické znázornění kritériální funkce MSE pro posun v osách x a y	40
4.5	Registrace dvojice snímků 27-28, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost	43
4.6	Q-Q graf výstupu registrace dvojice snímků 27-28, vlevo 2. nastavení optimalizace, vpravo 3. nastavení optimalizace	44
4.7	Graf časové náročnosti algoritmu PSO	45
4.8	Registrace dvojice snímků 27-28, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost	47
4.9	Q-Q graf výstupu registrace dvojice snímků 27-28, vlevo 2. nastavení optimalizace, vpravo 3. nastavení optimalizace	48
4.10	Graf časové náročnosti algoritmu BA	49
4.11	Registrace dvojice snímků 27-28, box plot vlevo: na ose x nastavení z 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost	51
4.12	Q-Q graf výstupu registrace dvojice snímků 27-28, vlevo 2. nastavení optimalizace, vpravo 3. nastavení optimalizace	52
4.13	Výpočetní náročnost DA	53
4.14	Registrace dvojice snímků 1-2	55
4.15	Registrace dvojice snímků 1-12	56
5.1	Ukázka snímků ze sekvence	57
5.2	Graf znázorňující měnící se průměrnou hodnotu jasu v sekvenci snímků, v závislosti na množství kontrastní látky	58
5.3	Průběh MSE napříč sekvencí pro vybrané algoritmy s 1. nastavením	59
5.4	Průběh MSE napříč sekvencí pro vybrané algoritmy s 2. nastavením	60

5.5	Registrovaná dvojice obrazů 1-28	62
6.1	Ukázka snímku ze sekvence s vyznačenou oblastí zájmu	63
6.2	Průběh MSE napříč sekvencí pro vybrané algoritmy s 1. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu	64
6.3	Průběh MSE napříč sekvencí pro vybrané algoritmy s 2. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu	65
7.1	Ukázka snímku ze sekvence s vyznačenou oblastí zájmu	66
7.2	Průběh MSE napříč sekvencí pro vybrané algoritmy s 1. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu	67
7.3	Průběh MSE napříč sekvencí pro vybrané algoritmy s 2. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu	68
7.4	Registrovaná dvojice obrazů 1-38	69
A.1	Registrace dvojice snímků 1-2, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost	78
A.2	Registrace dvojice snímků 1-13, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost	79
A.3	Registrace dvojice snímků 1-28, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost	79
A.4	Registrace dvojice snímků 12-13, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost	80
A.5	Registrace dvojice snímků 1-2, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost	80
A.6	Registrace dvojice snímků 1-13, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost	81
A.7	Registrace dvojice snímků 1-28, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost	81
A.8	Registrace dvojice snímků 12-13, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost	82
A.9	Registrace dvojice snímků 1-2, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost	82

- A.10 Registrace dvojice snímků 1-13, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost . 83
- A.11 Registrace dvojice snímků 1-28, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost . 83
- A.12 Registrace dvojice snímků 12-13, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost 84

Úvod

V poslední době roste zájem o aplikaci registrace obrazů v oblasti lékařské diagnostiky. Obraz získaný registrací, ať z jedné zobrazovací techniky či multimodální registrace, nese vyšší podíl informace a tudíž vede ke zvýšení kvality klinických snímků. S tím roste i přesnost diagnostiky a hodnocení zdravotních problémů.

Cílem registrace je nalezení takových parametrů transformace, které povedou k zarovnání snímků na sebe. K nalezení nejlepších hodnot parametrů se využívají optimalizační algoritmy. Kvalita transformace je popsána metrikou podobnosti mezi modelovým a transformovaným obrazem, která je během procesu optimalizace maximalizována. Díky snadnému konceptu a implementaci se k nalezení těchto parametrů využívá genetických algoritmů. Jejich metody jsou schopné čelit kombinaci optimalizačních problémů a výsledek není závislý na počátečním řešení, což dělá tento přístup robustním. Navíc jsou schopné uniknout z lokálních minim a dokáží pracovat i s multimodálními funkcemi.

Tato práce se věnuje monomodální registraci ultrazvukových snímků. Jelikož je vyšetření ultrazvukem neinvazivní, bezbolestné a bez negativních dopadů na lidský organismus, jeho potenciál je v oblasti medicínského zobrazování obrovský. Ultrazvukové zobrazování vytváří nepřetržitý proud obrazů a je tedy možné pozorovat anatomické struktury v reálném čase. Vysoký podíl šumové složky v ultrazvukových obrazech dělá jejich registraci těžší, než je například registrace dat z CT nebo MRI. Proto není registrace čistě ultrazvukových dat zcela běžně, prakticky používána a je zatím hlavně cílem studií.

Tato diplomová práce je rozdělena do sedmi základních kapitol. V první a druhé části je teoreticky popsán postup registrace obrazů, výčet metod genetických algoritmů a princip jejich fungování. Třetí kapitola je zaměřena na problematiku registrace, konkrétně ultrazvukových sekvencí. Jsou v ní zmíněny různé přístupy k registraci, jejich stručný popis a postupy u perfúzního zobrazování. Zbylé kapitoly obsahují popis praktické aplikace optimalizačních algoritmů roje částic (PSO), včelího algoritmu (BA) a algoritmu vážek (DA) pro registraci několika sekvencí ultrazvukových dat.

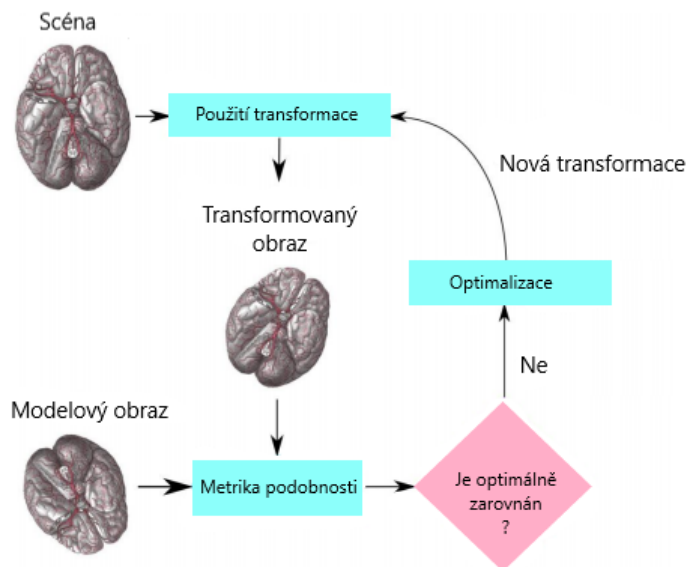
1 Registrace obrazu

Registrace obrazu je proces, jejímž cílem je zarovnat na sebe dva či více obrazů, které byly pořízeny za různých podmínek: různý časový okamžik, odlišné senzory, snímek pořízený z různých úhlů pohledu či kombinací výše zmíněných [11].

Zarovnání sady obrazových dat je popsáno geometrickými transformacemi. Snažíme se nastavit jejich parametry tak, abychom došli k co nejlepšímu výsledku registrace. K nalezení takového výsledku budeme dále používat tzv. optimalizační algoritmy.

1.1 Postup při registraci obrazu

Pro přehlednost registrace je třeba nejprve popsat jednotlivé části procesu. Vstupní obraz, nazývaný **scéna**, bude transformován k referenčnímu obrazu tzv. **modelu**. Míra podobnosti scény a modelu je popsána **metrikou podobnosti**. Odhad transformace je obvykle řešen iteračním **optimalizačním procesem**, kde se hledají její nejvhodnější parametry. Na obrázku 1.1 je tento proces graficky znázorněn.



Obr. 1.1: Optimalizační proces registrace obrazu, přeloženo z [1]

Cílem registrace obrazu (dále jen RO) je najít geometrickou transformaci T , která zarovná scénu I_S k modelu I_M tak, že transformovaná scéna $T(I_S)$ a model si budou co nejvíce podobné [1], tj.

$$\operatorname{argmin}_T \operatorname{Similarity}(I_M, T(I_S)). \quad (1.1)$$

Metoda RO se skládá ze tří základních procesů, a to je nalezení vhodné transformace k propojení obrazů, výpočet metriky podobnosti, která popisuje kvalitu zarovnání a optimalizační proces, jež zajistí nejlepší nastavení zvolené transformace [1].

1.2 Povaha obrazu

Vzhledem k povaze obrazu, musíme metody RO rozdělit na dvě základní skupiny a to voxel-based (intensity-based) a feature-based metody [1]. Tyto metody se používají před samotnou registrací. Slouží k extrakci redukované podmnožiny s nejvíce relevantními příznaky.

1.2.1 Voxel-based (intensity-based)

Tato metoda je založena na hledání odpovídajících vzorů ve scéně a modelu. Kvůli velkému množství dat je zapotřebí vymezit region, který bude prohledáván. Prohledávanou oblastí bývá zpravidla obdélníkové okno [2], které hledá shody mezi scénou a modelem. Takto zvolené okno má však své nevýhody, které se projeví, bude-li obraz deformován komplexní transformací. Okno nebude schopno pokrýt stejné části transformované scény a modelového obrazu. Další nevýhoda těchto metod se skrývá v obsahu obdélníkového okna. Tam, kde je vysoká pravděpodobnost, že okno obsahuje hladkou oblast, bez jakékoli podobnosti, bude nesprávně porovnán s jinými hladkými oblastmi v referenčním obraze. Proto je důležité, aby funkce pro registraci byly nejlépe využity ve výrazné části obrazu a je vhodné proto kontrolovat obsah okna. Ovšem hlavní nedostatky voxel-based metody se projeví při změně světelných podmínek akvizice těchto obrazů. V takovém případě budou podobnostní metriky nespolehlivé a optimalizační proces sklouzne do lokálního minima [3].

Přestože jsou tyto metody výpočetně složité, dosahují vysoké úrovně přesnosti a staly se za poslední desetiletí první volbou v analýze medicínských obrazů.

1.2.2 Feature-based

V této metodě se provádí zarovnání obrazů pouze na základě charakteristických částí v něm. Takové části se nazývají primitiva (features). Primitiva mohou tvořit čáry, body či regionální části obrazu [2]. Výhodou této metody je snížení složitosti problému, která se odvíjí od ruční či automatické volby primitiv. Při výběru primitiv je důležité, aby nesly informaci, která charakterizuje obraz [1]. Voxel-based metody jsou schopny pracovat s větším množstvím dat, zatímco u feature-based metod dojdeme k hrubšímu odhadu globálního řešení z redukovaného množství dat.

1.3 Transformace scény

Transformace scény popisuje geometrickou transformaci, která byla použita k získání modelového obrazu. Geometrické transformace můžeme dle [4] rozdělit na dvě základní skupiny a to na rigidní a elastické. Rigidní transformace nemění vnitřní geometrii obrazu, ale pouze posouvají či otáčejí celý obraz. Za to u elastických transformací dochází k jisté geometrické deformaci [9].

Každá z těchto skupin má určité charakteristické stupně volnosti, které představují počet nezávislých parametrů a jsou nutné k určení transformace. Maticové transformace můžeme dle článku [18] rozdělit na čtyři základní typy: striktně rigidní, podobnostní, afinní a perspektivní. Všechny tyto maticové transformace mohou provádět translaci a rotaci. Podobnostní transformace popisují škálování a afinní transformace postihují všechny výše zmíněné transformace, a navíc i škálování. Takzvané perspektivní transformace, nevyžadují zachování rovnoběžnosti hran v obraze a mají nejvíce stupňů volnosti. Elastické (ne – rigidní) transformace mohou mít mnohem více stupňů volnosti [1] a mohou tak modelovat pružné deformace anatomických předmětů. Elastické transformace s více parametry tedy často vyžadují regularizaci, aby byl udržen prostorový řád a pravidelnost.

Výběr transformačního modelu zcela závisí na konkrétní aplikaci. Při běžném snímání může stačit výpočet translace. Ovšem například při snímání dýchacích pohybů je zapotřebí počítat i s lokálními deformacemi. I když se podařilo vytvořit flexibilní transformační model, neznamená to, že proces bude úspěšný. Nejen, že bude výpočetně náročný, ale také pravděpodobně povede k anatomicky nežádoucím zkreslením, jako například k ohybu kostí či tkání [1].

1.3.1 Rigidní transformace

Tato lineární transformace popisuje posun a rotaci obrazu. Nejjednodušším případem je pouhý posun [4]

$$\Delta r = [\Delta x, \Delta y]^T, \quad (1.2)$$

jehož zjednodušená transformace

$$r' = r + \Delta r, \quad (1.3)$$

má dva parametry, a to prostorové posuny podél jednotlivých souřadnic. Rotace je popsána transformační maticí B [4], která je ve 2D prostoru definována takto

$$B = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (1.4)$$

dochází tak k otáčení bodů v rovině xy proti směru hodinových ručiček o úhel θ vzhledem k ose x kolem počátku soustavy souřadnic. Rotace v 3D prostoru je definována třemi rotačními úhly $\theta_x, \theta_y, \theta_z$ kolem příslušných souřadnicových os. Výsledná transformační matice se pak skládá z postupných rotací v jednotlivých osách

$$B = \begin{pmatrix} \cos \theta_z & -\sin \theta_z & 0 \\ \sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \theta_y & 0 & \sin \theta_y \\ 1 & 1 & 0 \\ -\sin \theta_y & 0 & \cos \theta_y \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x \\ 0 & \sin \theta_x & \cos \theta_x \end{pmatrix}. \quad (1.5)$$

Obecná rigidní transformace obsahuje tedy obě již výše popsané složky [4]

$$r' = Br + \Delta r, \quad (1.6)$$

kdy ve dvourozměrném prostoru by měla transformace tři parametry a v třírozměrném šest. Výhodou rigidní transformace je to, že se dá zapsat pomocí jediné translačně-rotací matice [4]

$$r' = Rr, \quad (1.7)$$

a to ve 2D interpretaci

$$r' = \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta & \Delta x \\ \sin \theta & \cos \theta & \Delta y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (1.8)$$

a ve 3D interpretaci (při označení $B = [b_{ik}]$)

$$r' = \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} b_{11} & b_{12} & b_{13} & \Delta x \\ b_{21} & b_{22} & b_{23} & \Delta y \\ b_{31} & b_{32} & b_{33} & \Delta z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}. \quad (1.9)$$

1.3.2 Elastická transformace

Elastickou transformaci je možné si představit jako obraz vytištěný na pružném podkladu, jehož části mohou být roztahovány nebo stlačovány. Nejjednodušší transformací tohoto druhu je změna měřítka[4]

$$r' = Sr, \quad (1.10)$$

kde pro 2D a 3D

$$S = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad S = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_y & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}. \quad (1.11)$$

Budou-li všechny hodnoty s_i totožné, dojde pouze ke zvětšení nebo zmenšení obrazu. Jinak dojde ke změně proporcí v obraze. Další pružnou transformací je tzv. zkosení[4], které je definováno ve dvourozměrném prostoru jako

$$G = G_x G_y, \quad G_x = \begin{pmatrix} 1 & g_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad G_y = \begin{pmatrix} 1 & 0 & 0 \\ g_{yx} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1.12)$$

a třírozměrnému prostoru

$$G = G_x G_y G_z, \quad (1.13)$$

$$G_x = \begin{pmatrix} 1 & g_{xy} & g_{xz} & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad G_y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ g_{yx} & 1 & g_{yz} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad G_z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ g_{zx} & g_{zy} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (1.14)$$

Nejobecnější lineární transformací obrazu je tzv. afinní transformace[4], která sdružuje dohromady translačně-rotační matici, změnu měřítka i zkosení

$$r' = GSR_r = A_r, \quad (1.15)$$

kde matice dvou a tří dimenzionálních dat budou vypadat takto

$$A = \begin{pmatrix} a_{11} & a_{12} & t_1 \\ a_{21} & a_{22} & t_2 \\ 0 & a_{22} & 1 \\ 0 & 0 & 1 \end{pmatrix}, \quad A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ a_{31} & a_{32} & a_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (1.16)$$

Afinní transformace má 6 nezávislých parametrů pro 2 rozměry, nebo 12 nezávislých parametrů pro 3 rozměry a zachovává rovnoběžnost hran.

1.3.3 Lokální transformace

K registraci globálních transformací se využívá lineární registrace, kde se k výpočtu parametrů transformační funkce zužikovává celý obrazový prostor. Ovšem tento přístup nemusí být dostatečný, zabýváme-li se oblastí lékařského zobrazování. Pohyb pacienta, či přímo pohyb snímacího zařízení, může vést k lokálním deformacím scény. Globální lineární registrace je tak rozšířena na nelineární registraci popisující lokální deformace. Parametry, které registrace odhaduje prostřednictvím optimalizační strategie, odpovídají tak stupňům volnosti modelu deformace a jejich počet se velmi liší. V případě globálních tuhých transformací, dosáhneme 6 parametrů a v případě neparametrických hustých transformací jich mohou být až miliony [17].

Nelineární registrace patří do skupiny elastických a jejím hlavním problémem je správné přiřazení kontrolních bodů a volba mapovací funkce pro registraci. Protože meziobrazové deformace mohou být různé a zvláště u lékařských zobrazení je problém rozlišit deformace obrazu od skutečné změny ve scéně, nelze využít žádného automatického algoritmu. Obecným postupem nelineární transformace je nalezení takové prostorové funkce, která namapuje všechny body ve scéně na jejich odpovídající body v referenčním obraze. Tímto získáme tzv. deformační mřížku, která reprezentuje vychýlení jednotlivých pixelů v 2D obraze.

Nelineární registrace dělíme na dvě základní skupiny, a to parametrické a neparametrické. Do neparametrických registrací se řadí registrace elastické (elastic registration) nebo viskoelastické (visco-elastic registration), nazývané též jako tekuté registrace. Fluidní mapování zaručuje, že se scéna hladce deformuje, aniž by se roztrhla, i když jsou pro přesné přizpůsobení vyžadovány velké deformace [16]. Tento způsob registrace je hojně využíváný v lékařství, kde se používá ke sledování anatomických změn, k opravám zkrasleného obrazu či integraci multimodálních dat. Druhou skupinou jsou parametrické registrace využívající algoritmu nejmenších čtverců či interpolačních metod [2]. Interpolace je realizována konvolucí obrazu s interpolačním jádrem. Nejjednodušší metodou interpolace je metoda nejbližšího souseda (nearest neighbor) [4]. Hledanou hodnotu zde určíme jako hodnotu intenzity nejbližšího známého bodu. Další oblíbenou metodou je interpolace pomocí B - splajnů, kde uzly mřížky mohou sloužit jako řídicí body splajnu [9]. Každý uzel v mřížce je iterativně translatován s cílem optimalizovat danou kritériální funkci. Výhodou B - splajnu je jeho lokální působení, které umožňuje vyhodnocení v každé jednotlivé oblasti zájmu, což znamená, že v každé iteraci je transformována pouze malá oblast obrazu, i když je míra podobnosti vyhodnocována přes celý obrázek. Metod interpolace je celá řada. Mezi další využívané patří například funkce sinc, lineární metody či interpolace Lagrangeovými polynomy.

1.4 Strategie prohledávání optimalizačního procesu

Jak již bylo zmíněno, cílem RO je najít vhodnou transformační funkci, která zarovná sadu obrazů do jednoho souřadnicového systému. Podle strategie vyhledávání potřebných komponent můžeme rozlišit dva základní přístupy. Prvním z nich je matching-based přístup, který řeší spojitosti mezi dvojicemi podobných znaků v obrazech. Zatímco v druhém přístupu, tzv. parameters-based je přímo prohledáván prostor s parametry funkce. Obě tyto metody nelze vyřešit přímou metodou, jelikož tu vzniká nelineární optimalizační problém. Odhad funkce je tak hledán postupně v iteracích, kdy lze použít klasické numerické optimalizátory. Nicméně řešení většinou končí v lokálních minimech [3].

1.4.1 Matching-based RO

Jedná se o optimalizační proces, který prohledává prostor geometrických transformací a hledá společné znaky či regiony v obrazech. Prvním krokem je stanovení počtu společných rysů mezi scénou a modelem, což bývá řešeno za pomoci evolučních algoritmů či metaheuristiky. Druhým krokem je určit optimální transformační funkci odpovídající předchozímu kroku. Nejběžněji používanou numerickou metodou je výpočet Least squares estimators. A to díky tomu, že počítají pouze s průměry, variancemi a kovariancemi.

Metody založené na porovnávání používají algoritmy jako Iterative Closest Point (ICP) [5]. Je to iterativní gradientní metoda, kde funkce F (běžně MSE) hraje roli pouze jakéhosi ukončovacího kritéria. Odhad transformace je závislý na výstupu z procesu hledání společných parametrů. Tudíž, čím lepší je výběr shodných parametrů, tím přesnější je odhad transformace. V důsledku toho je hodnota metriky podobnosti přesnější. ICP algoritmus má však i své nevýhody. Dá se použít pouze na normálně distribuovaná pozorování, je závislý na počátečním odhadu a často uvízne v lokálním optimu. A na konec jeden z obrazů (běžně scéna) by měl být obsažen v tom druhém [3].

1.4.2 Parameters-based RO

Metoda generuje možné vektory hodnot parametrů, které jsou uchazeči o registrační transformace. V této metodě hledáme řešení v prostoru parametrů [1]. Každá registrační transformace je tvořena vektorem parametrů, mezi kterými se v průběhu procesu vybírá ta nejlepší. Na rozdíl od ICP metod, je prohledávání prostoru tvořeno přímo metrikou podobnosti. Tento problém numerické optimalizace obsahuje hledání nejlepších parametrů, které minimalizují F .

Rozhodující pro tuto metodu RO jsou řády v měřítku parametrů. Jednotlivé změny úhlu mají na obraz větší dopad než translační změny. Rozdíly v měřítku se projeví jako protáhlá údolí v prostoru pro vyhledávání parametrů, což působí potíže pro klasické gradientní optimalizátory. Konvergence není zaručena a může ve většině případů skončit v lokálních minimech [3].

1.5 Metrika podobnosti

Metrika podobnosti vyjadřuje podobnost dvou obrazů. To, jak moc jsou si transformovaná scéna a model podobné je popsáno různými funkcemi. Zvolená metrika závisí na původu obrazu a přístupu k registraci. U feature-based metod se obvykle využívá výpočtu vzdáleností mezi prvky modelu x_i a transformované scény $T(x_i)$. Takové zarovnání může být hodnoceno pomocí metriky střední kvadratické odchylky (MSE) [1]

$$MSE = \frac{1}{r} \sum_{i=1}^r |x_i - c(T(x_i))|^2. \quad (1.17)$$

V intensity-based metodách se využívá podobnosti intenzit jasu pixelů ve dvou obrazech. Pokud bychom předpokládali, že je tento vztah lineární, pro jejich podobnost bychom mohli zvolit výpočet lineárního korelačního koeficientu. Tato metrika se nazývá normalizovaná korelace (NC) [4], která je definovaná takto

$$NC(a, b) = \frac{\sum_i (a_i - \bar{a}_i)(b_i - \bar{b}_i)}{\sqrt{\sum_i (a_i - \bar{a}_i)^2} \sqrt{\sum_i (b_i - \bar{b}_i)^2}}, \quad \bar{a} = \frac{1}{N} \sum_i a_i, \quad \bar{b} = \frac{1}{N} \sum_i b_i, \quad (1.18)$$

kde \bar{a}_i a \bar{b}_i jsou průměrné hodnoty intenzity obrazu. Tato míra podobnosti je počítána pro dvojici oken snímaného a referenčního obrazu a hledá se jejich maximum. Pokud požadujeme přesnost registrace subpixelů, je třeba spočítat interpolaci naměřených hodnot NC [9]. Metrika korelace může být využita i u mírně pootočených obrazů a obrazů, které prošly mírnou změnou měřítka [2]. Rychlejší ovšem méně přesnou alternativou metriky NC je algoritmus SSDA (Sequential Similarity Detection Algorithm). Shromažďuje sumy absolutních hodnot rozdílů intenzit pixelů a aplikuje na ně prahové kritérium. Pokud kumulovaný součet překročí danou hranici, dvojice kandidátských oken je odmítnuta a je testován další pár referenčního a snímaného okna. SSDA používá sekvenční vyhledávací přístup a výpočet vzdáleností je jednodušší, než u NC [2].

V této době je velký zájem o multimodální registraci, která je založena na poměru korelací [10]. Na rozdíl od klasické NC dokáže pracovat s rozdílnou intenzitou obrazů, které byly získány z odlišných senzorů. Právě intenzita je tu reprezentována funkcí. Obecně však metody založené na korelaci mají dvě hlavní nevýhody a tj. plochost maxima míry podobnosti (kvůli podobnosti obrazů) a vysoká výpočetní

náročnost. Maximum může být vyzdviženo předzpracováním obrazů či za pomoci hranové či vektorové korelace. Ovšem nevýhodou je, že metoda na takto předzpracovaný obraz je méně citlivá na rozdíly v intenzitě mezi referenčním a snímaným obrazem. Navzdory výše zmíněným omezením se stále běžně používají metody korelace podobné registraci. Jsou snadno implementovatelné a umožňují aplikaci v reálném čase.

Další metrikou, která je jednou z nejvíce žádaných ze skupiny regionálních metod je metoda vzájemné informace (MI). Má vysoké uplatnění v tzv. multimodální registraci obrazu. Registrace dvou a více obrazů, které byly nasnímány odlišnými modalitami, může být silně nelineární. MI pochází z informační teorie a vyjadřuje statistickou závislost mezi dvěma soubory dat [2]. Je definována takto

$$MI(I_A, I_B) = \sum_{a \in I_A} \sum_{b \in I_B} p_{AB(a,b)} \log \frac{p_{AB(a,b)}}{p_A(a)p_B(b)}, \quad (1.19)$$

kde p_{AB} a p_A, p_B jsou společné a marginální pravděpodobnosti distribuce hodnot intenzity v obrázcích [1].

Metrika podobnosti je považována za hlavní součást RO, jelikož nám vyhodnocuje výsledky geometrické transformace obrazů [1].

2 Metody optimalizace

Metaheuristika (MH) nebo spíše heuristika je optimalizační proces, který efektivně prohledává prostor a nachází to nejlepší řešení. Řeší náročné procesy optimalizace a stačí jen malá změna a algoritmus je schopen se adaptovat na specifické problémy. Obecné HM algoritmy se využívají napříč nejrozličnějším oborům, jako jsou kromě registrace medicínských obrazů například ekonomika či průmysl. Mezi takové algoritmy patří například algoritmus simulovaného žíhání (SA), tabu prohledávání (TS), Particle swarm optimalizace (PSO), Scatter search (SS) a další [1].

Do rodiny MH patří často využívaná skupina evolučních algoritmů (EA). Tyto modely evolučních algoritmů se inspirovaly v přírodě a představují procesy vyvíjející se populace. Takové algoritmy jsou schopny dosáhnout dobrých výsledků při hledání globálního optima. Mezi základní modely patří genetický algoritmus (GA), evoluční strategie (ES), genetické programování (GP) a evoluční programování (EP).

Díky snadnému konceptu a implementaci se evoluční, potažmo metaheuristické algoritmy jeví jako vhodná volba pro registraci obrazů. Jsou schopny čelit kombinaci optimalizačních problémů, a to transformaci parametrů a matching based přístupům. Oproti gradientním metodám výsledek EA není závislý na počátečním řešení, což tento přístup dělá více robustním. Navíc obsahuje i konkrétní strategie, jak uprchnout z lokálních minim a dokáže pracovat i s multimodálními funkcemi. Tyto algoritmy fungují i s libovolnými druhy omezení. Zpracují konkrétní požadavky a vymezí široký rozsah možných cílů. Obecně jsou EA a MH časově náročné a neexistuje žádný důkaz, že tyto přístupy konvergují do globálního optima. Avšak spolu s metodami založenými na gradientu mohou vytvořit hybridní strategii k řešení problému a jsou schopny utéct z lokálních minim a dostat se ke globálnímu řešení [3].

2.1 Metaheuristické a evoluční algoritmy

2.1.1 Simulované žíhání

Jedná se o metodu, která se inspirovala technikou využívanou v metalurgii, kdy zahřejeme krystal na vysokou teplotu a pomalu jej ochlazujeme (žíháme) za vzniku stabilní krystalické mřížky s minimem defektů. Tento pravděpodobnostní optimalizační algoritmus je schopen přijímat i horší řešení a uniknout tak z lokálních minim [1]. V každé iteraci algoritmus zvažuje řešení v sousedství $f(y)$ toho současného $f(x)$ a spočítá jeho kvalitu. Pokud je tato kvalita vyšší, stane se aktuálním řešením, teplotu snižujeme a iterace končí. Pokud je horší, pak přijímáme řešení s určitou

pravděpodobností a snížíme teplotu T dle ochlazovacího plánu [5].

$$p(x, y) = e^{\frac{f(y) - f(x)}{T}} \quad (2.1)$$

Počáteční teplotu nastavíme na vysokou hodnotu a dále ji v iteracích snižujeme lineárně nebo exponenciálně. Při vysokých teplotách bude řešení přijato bez ohledu na jeho kvalitu a algoritmus bude připomínat náhodné prohledávání. Při teplotách blízkých nule budou naopak horší řešení přijímána zřídka. Čím pomaleji teplota klesá, tím více algoritmus prohledává prostor. Optimalizační proces nemusí konvergovat ke globálně optimálnímu řešení. Lze tedy stanovit ukončovací kritéria algoritmu jako jsou například stanovení minimální teploty, poměr zamítnutých řešení či rozptyl navštívených řešení [6].

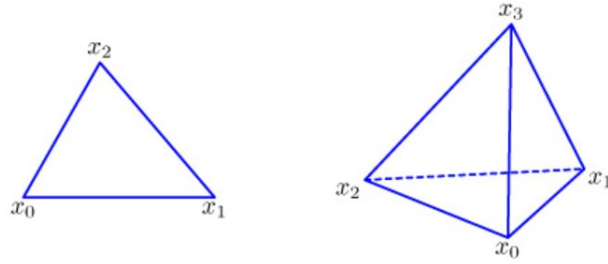
2.1.2 Zakázané prohledávání

Zakázané prohledávání, tzv. Tabu prohledávání, vychází z horolezeckého algoritmu, který se snaží odstranit problém zacyklení v lokálním optimálním řešení. Obě metody vycházejí z náhodně navrženého řešení, které se stane středem pro nové okolí, kde hledáme minimum [29]. Základní nevýhodou prostého horolezeckého algoritmu je to, že po určitém počtu iterací má metoda tendenci vracet se do lokálního optimálního řešení, které se již vyskytlo v předcházejícím kroku. Tento problém se snaží odstranit algoritmus zakázaného prohledávání, kdy jeho hlavní myšlenkou je vytvořit tzv. tabu list [29] – krátkodobou paměť, která dočasně zapisuje inverzní transformace k transformacím použitým v předcházejících iteracích. Tyto inverzní transformace jsou zakázány při tvorbě nového okolí pro dané aktuální řešení. Tabu list je při inicializaci algoritmu prázdný a postupně se plní. Až je plný, tak se v něm nejstarší hodnoty přepisují. Důležitým parametrem je velikost tabu listu, protože jeho délka ovlivňuje možnost vymanit se z lokálních minim. Když bude seznam krátký, může dojít k zacyklení algoritmu stejně jako u horolezeckého. Bude-li seznam dlouhý, můžeme minout lokální minima, která mohou být minimy globálními. Oblíbenou úpravou prohledávání je zavedení tzv. dlouhodobé paměti. Algoritmus se tak adaptuje na délku zakázaného seznamu a do dlouhodobé paměti zapisuje nejčastěji používané transformace, které se postupně penalizují, aby dostaly šanci i jiné transformace. Metoda zakázaného prohledávání může být použita ve spojení s horolezeckým algoritmem, ale i v kombinaci s jinými algoritmy, jako jsou např. genetické či simulované žíhání.

2.1.3 Nelder-Meadův algoritmus

Nelder-Meadův algoritmus nebo jinak simplexová metoda je jedním z nejpoužívanějších algoritmů pro multidimenzionální nepodmíněnou optimalizaci [43]. Základem je simplex S , kterým je v případě dvojrozměrného prostoru trojúhelník, ve třech dimenzích čtyřstěn a je zobrazen na obrázku 2.1. Je tvořen množinou nekomplanárních bodů $d + 1$ v D prostoru jako

$$S = \{x_1, x_2, \dots, x_{d+1}\}. \quad (2.2)$$



Obr. 2.1: Představa Simplexu, převzato z [43]

V simplexu nalezneme bod s nejnižší x_L a nevyšší x_H funkční hodnotou [44]

$$x_H = \operatorname{argmax}_{x \in S} f(x) \quad (2.3)$$

$$x_L = \operatorname{argmin}_{x \in S} f(x) \quad (2.4)$$

a spočítáme těžiště g jako průměr d bodů:

$$g = \frac{1}{d} \sum_{x \in S} x - x_H. \quad (2.5)$$

V iteracích se hledá nový bod y , který nahradí dosud nejhorší bod simplexu s hodnotou x_H . Budeli splněna podmínka $f(y) < f(x_H)$, nový bod se hledá překlopem x_H přes těžiště tzv. **reflexí** dle rovnice:

$$x_R = g + (g - x_H) \quad (2.6)$$

a hodnota v něm je $f_R = f(x_R)$. Jestliže funkční hodnota bodu x_R leží mezi funkčními hodnotami x_L a x_S , pak ji přijímáme. Jinak počítáme **expansi**, kde poloha tohoto bodu bude:

$$x_E = g + \gamma(x_R - g), \quad (2.7)$$

kde γ je expanzní faktor určující míru expanze [45] a hodnota v něm je $f_E = f(x_E)$. Jestliže $f_S \leq f_R$, pak nastane **kontrakce** x_C jako jedna z následujících možností. Pokud $f_S \leq f_R \leq f_H$, pak

$$x_{C1} = g + \beta(x_R - g), \quad (2.8)$$

kde β je kontrakční faktor [45] a $f_{C1} = f(x_{C1})$. Pokud $f_{C1} < f_R$, pak je akceptována x_{C1} . Jinak dochází k redukci. Ale pokud $f_H \leq f_R$, pak je bo kontrakce počítán

$$x_{C2} = g + \beta(x_H - g), \quad (2.9)$$

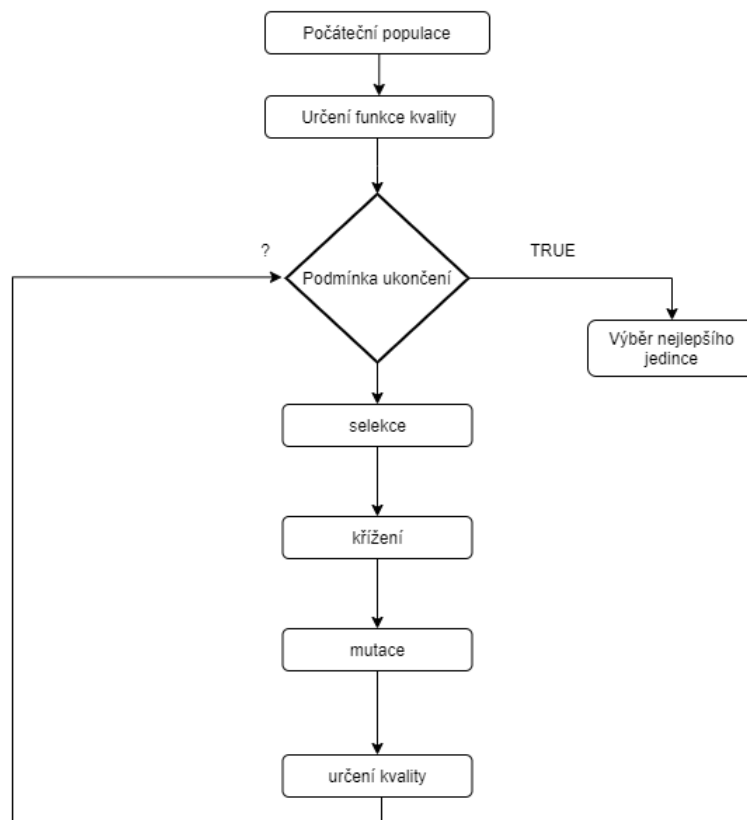
kde $f_{C2} = f(x_{C2})$. Pokud $f_{C2} \leq f_H$, pak je tento bod přijat, jinak je počítána redukce. **Redukce** je počítána jako d nových vrcholů x_j dle vzorce 2.10 pro $j=0,1,\dots,d$, když $j \neq L$.

$$x_j = x_L + \Delta(x_j - x_L). \quad (2.10)$$

Nejhorší řešení simplexu se v algoritmu stále transformují do splnění ukončovací podmínky, kterou bývá konvergence vrcholů simplexu k jednomu bodu či zvolený počet iterací [44].

2.1.4 Genetický algoritmus

Připomíná evoluční proces, ve kterém se v iteracích vyvíjí lepší a vhodnější řešení problému. Máme populaci jedinců, která prochází evolucí. Každý jedinec představuje jedno řešení daného problému. Jednotlivci jsou zapsáni ve formě chromozomu [11]. Každá pozice v chromozomu se nazývá gen a hodnotě genu se říká alela. Chromozomální kódování výsledku se nazývá genotyp a samotné kódování vlastností se označuje jako fenotyp jedince. V genetických algoritmech (dále GA) se používá převážně binární kódování vlastností jedinců. V první generaci, je populace složena z náhodných členů. Při přechodu jedinců do další generace je vypočítána pro každého z nich tzv. funkce kvality, která vyjadřuje kvalitu řešení. Ti nejlepší jedinci, dle hodnoty funkce kvality, jsou náhodně vybráni a dále podstupují mutaci a křížení, z čehož vznikne nová populace. Tento postup se iterativně opakuje, dokud není dosaženo ukončovacího pravidla, jako je například maximální počet generací. Postup GA je znázorněn ve vývojovém diagramu na obrázku 2.2.



Obr. 2.2: Vývojový diagram genetického algoritmu

K tvorbě „potomků“ je nejprve třeba zvolit „rodiče“. Dle Darwinovy teorie o původu druhu, silnější jedinci mají větší šanci se prosadit a žít déle. K tzv. selekci rodičů může dojít náhodným výběrem, kdy není nikterak zohledněna funkce kvality jedinců. Nebo se využívají dvě základní metody, a to metoda rulety a turnajové selekce [6].

2.1.5 Particle Swarm Optimalizace (PSO)

Na rozdíl od genetických algoritmů, kde vznikali během evoluce noví jedinci, PSO pouze zlepšuje stávající jedince pomocí jejich pohybu. Algoritmus je inspirován sociálními aspekty, jako jsou například rojení hmyzu či chování hejna ptáků [7]. Jedná se o stochastický algoritmus, pracující s náhodně rozmístěnými jedinci v populaci. Tito jedinci se nazývají částice a mají tendenci se pohybovat a prohledávat okolní prostor, případně se shlukovat v oblasti minim.

Pro každou částici je vygenerován vektor rychlosti (v_i), udávající směr pohybu částice v příštím kroku. Pohyb jedince je tvořen třemi složkami. Individuální (v_i), kdy částice pokračuje svou cestou, konzervativní (p_i), částice se pohybuje ke své

dosud nejlepší pozici a přizpůsobivou (g), kdy se pohybuje k pozici dosud nejlepšího jedince. Rychlost a pozici částice lze popsat vzorcem [7]

$$v_i(t) = w(t)v_i(t-1) + \varphi_1 u_1(p_i - x_i(t-1)) + \varphi_2 u_2(g - x_i(t-1)) \quad (2.11)$$

$$x_i(t+1) = x_i(t) + v_i(t), \quad (2.12)$$

kde $w(t)$ jsou vnitřní váhy, φ je konstanta zrychlení a $u_{1,2} \in (0, 1)$ jsou rovnoměrně rozložená náhodná čísla.

Metoda PSO je závislá na počátečním nastavení vah $w(t)$, konstantě zrychlení, maximálním počtu iterací, ale také na počátečním nastavení populace [8]. Její velkou výhodou je snadná implementace a rychlá konvergence k optimálnímu řešení. Získala tak na popularitě při řešení různých optimalizačních úloh. Na obrázku 2.3 je zobrazen její stručný pseudokód.

- 1: Náhodná inicializace počáteční populace částic
- 2: Generování vektoru rychlosti udávající směr pohybu v příštím kroku pro každou částici
- 3: **while** není splněna podmínka **do**
- 4: Výpočet fitness populace
- 5: Aktualizace fitness hodnoty a pozice nejkvalitnější částice z celé populace a aktualizace dosavadní nejlepší hodnoty a pozice každé částice
- 6: Aktualizace rychlosti jedince dle rovnice 2.11
- 7: Aktualizace pozice jedince dle rovnice 2.12
- 8: **end while**

Obr. 2.3: Pseudokód PSO algoritmu

2.1.6 Včelí algoritmus (BA)

Dalším ze skupiny metod inspirovaných přírodou je včelí algoritmus (BA). Tato rojová inteligence vychází z přirozeného chování včel a implementace algoritmu je jednoduchá a intuitivní. Algoritmus se skládá ze dvou skupin včel - průzkumnic a včel vyčkávajících [32].

Včely průzkumnice hledají kolem úlu květiny plné pylu. Létají náhodně z jednoho květu na druhý. Po návratu do úlu uloží nasbíraný nektar a ostatním včelám předvádí "taneční rituál", kterým informují ostatní včely o směru, výskytu, kvalitě a vzdálenosti květů. Z průzkumnic se na počátku každého iterativního kroku vybere m nejlepších včel, které se v závislosti na kvalitě svých řešení rozdělí na e elitních a $m - e$ ostatních vybraných včel. Dále z úlu vyrazí včely doposud vyčkáující, které

prohledávají m vybraných míst, kde elitní včela získá více následovnic, než včela neelitní, což umožňuje rychle a efektivně hledat zdroj potravy [31]. Včely následovnice lokálně prohledávají prostor okolo zvolené průzkumnice. V případě, že najdou lepší řešení, průzkumnice si jej aktualizuje. V další iteraci se pak bude generace průzkumnic skládat z m vybraných průzkumnic z předchozí iterace a $n - m$ nových průzkumnic, které jsou náhodně rozložené. Tyto kroky se opakují dokud není dosaženo ukončovacího kritéria [30]. Na obrázku 2.4 je znázorněn pseudokod základního algoritmu včel. Na konci každé iterace bude kolonie mít populaci složenou ze dvou částí: nejlepších včel, které jsou vybrány z jednotlivých oblastí a přidělených včel vyčkávajících, které provádí náhodné prohledávání.

Stejně jako jiné rojové algoritmy má i BA další varianty rozšíření. Nejzmiňovanější z nich je algoritmus umělé včelí kolonie (ABC) [33], kde je roj rozdělen na tři druhy včel - dělnice, vyčkávající a včely průzkumnice. Se zavedením včel dělnic, přichází do algoritmu další parametr - tzv. kritérium opuštění zdroje, kde pokud nedojde během několika iterací k nalezení potravy v okolí zvoleného zdroje, stává se z včely dělnice včela průzkumnice, která si vybere nějaký náhodný bod v prohledávaném prostoru a změní se opět v dělnici na této pozici.

- 1: Náhodná inicializace počáteční populace n včel průzkumnic
- 2: Výpočet fitness populace
- 3: **while** není splněna podmínka **do**
- 4: Výběr m nejkvalitnějších včel (lokalit)
- 5: Určení velikosti okolí lokalit
- 6: Náhodné umístění vyčkávajících včel do okolí (pro e elitních jedinců více než pro zbylých $m - e$) a výpočet jejich fitness
- 7: Výběr nejkvalitnějších jedinců z každého okolí
- 8: Doplnění $n - m$ náhodně rozložených průzkumnic a výpočet jejich fitness
- 9: **end while**

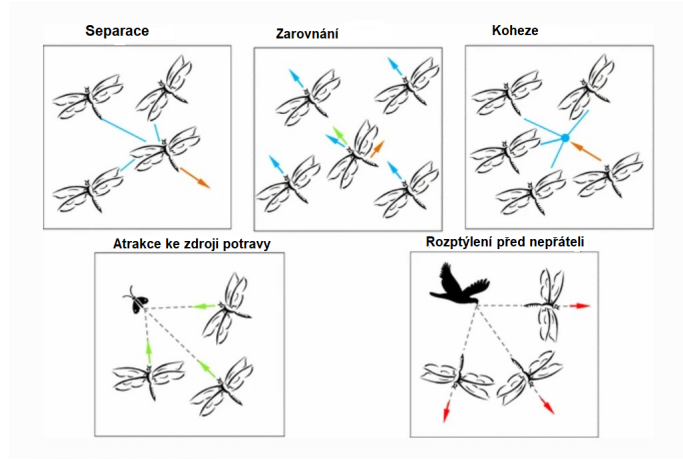
Obr. 2.4: Pseudokód BA algoritmu [40]

2.1.7 Dragonfly algoritmus (DA)

Vážky jsou drobní létající dravci, kteří se živí širokou škálou drobného hmyzu, motýly, včelami či mravenci. Na světě existuje 3000 druhů vážek, jejichž chováním se inspiroval v roce 2016 Mirjalili [35] a vytvořil tak nový optimalizační algoritmus, který zaujal mnoho výzkumníků v několika vědních oblastech.

Chování roje vážek se dá popsat statickou (lov kořisti) a dynamickou (migrace) složkou. Ve statické části vážky tvoří malé skupiny, které létají tam a zpět na malém uzemí a loví kořist. V dynamické fázi pak velké množství vážek migruje v jednom

směru na dlouhou vzdálenost a odvádí pozornost od nepřítele [36]. K modelování chování roje vážek uvedeného na obrázku 2.5 se využívá pěti základních primitivních principů, kterými dle [37] jsou separace, zarovnání, koheze, atrakce a rozptýlení.



Obr. 2.5: Vzory chování mezi vážkami v roji, upraveno z [35]

Každé z těchto principů chování lze dle Reynoldse [37] matematicky popsat.

Separace představuje zamezení statické kolize jednotlivců s ostatními v sousedství.

Matematicky je vyjádřena jako:

$$S_i = - \sum_{j=1}^N X - X_j, \quad (2.13)$$

kde X značí pozici konkrétního jednotlivce, X_j je pozice j -té sousední vážky, N je počet sousedů v roji.

Zarovnání označuje přizpůsobení rychlosti jednotlivce, mezi ostatními sousedy stejné skupiny. Zarovnání je znázorněno následovně:

$$A_i = \frac{\sum_{j=1}^N V_j}{N}, \quad (2.14)$$

kde v_j je rychlost j -tého jedince.

Koheze představuje soudržnost jednotlivců směrem ke středu roje a je matematicky popsána:

$$C_i = \frac{\sum_{j=1}^N X_j}{N} - X. \quad (2.15)$$

Atrakce, neboli přitažlivost ke zdroji potravy (F) je modelována jako:

$$F_i = X^+ - X, \quad (2.16)$$

kde X je pozice konkrétního jedince a X^+ značí pozici zdroje potravy.

Rozptýlení před nepřáteli je popsáno:

$$E_i = X^- + X, \quad (2.17)$$

kde X je pozice jedince a X^- je pozice nepřítele.

Chování vážek je definováno kombinací pěti výše zmíněných vzorů pohybu. Pozice simulované vážky uvnitř prohledávaného prostoru je popsána, podobně jako u algoritmu PSO, krokovým vektorem ΔX a pozicí X . Krokový vektor ukazuje směr pohybu vážek a je definován následovně:

$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + w\Delta X_t, \quad (2.18)$$

kde s je váha separace, S_i značí separaci i -tého jedince, a je váha zarovnání, A_i je zarovnání i -tého jedince, c je váha koheze a C_i je koheze i -tého jedince a f je faktor potravy, F_i značí zdroj potravy i -tého jedince a e je faktor nepřítele, E_i je pozice i -tého nepřítele, w je interní váha a t je čítač iterací.

Po výpočtu krokového vektoru ΔX je spočítán vektor pozice:

$$X_{t+1} = X_t + \Delta X_{t+1}, \quad (2.19)$$

kde t je konkrétní iterace.

Při prohledávání prostoru je váha zarovnání vysoká a koheze naopak nízká, zatímco u vykořisťování je to opačně. Konvergence algoritmu je kontrolována adaptivním přizpůsobováním vah, kdy je zaručené sblížování jedinců během procesu optimalizace [38]. V konečné fázi se roj stává jednou skupinou, která konverguje ke globálnímu optimu. Nejlepší a nejhorší dosud nalezená řešení se stávají zdrojem potravy a nepřáteli.

Aby bylo zlepšeno stochastické chování vážek a jejich prohledávání prostoru, je zaveden tzv. Lévyho let [39] pro případ, kdy neexistují žádná sousední řešení. V takovém případě se poloha vážky aktualizuje dle následující rovnice:

$$\Delta X_{t+1} = X_t + Lévy(d) \times \Delta X_t, \quad (2.20)$$

kde t je konkrétní iterace a d je dimenze pozičního vektoru.

Lévyho let se dle [35] vypočítá :

$$Lévy(x) = 0.01 \times \frac{r_1 \times \sigma}{|r_2|^{\frac{1}{\beta}}}, \quad (2.21)$$

kde r_1, r_2 jsou dva náhodné vektory rovnoměrně rozložené v rozsahu $[0,1]$, β je konstanta a σ se spočítá takto:

$$\sigma = \left(\frac{\Gamma(1 + \beta) \times \sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2}) \times \beta \times 2^{(\frac{\beta-1}{2})}} \right)^{\frac{1}{\beta}}, \quad (2.22)$$

kde $\Gamma(x) = (x - 1)!$.

Pseudokód k algoritmu DA je uveden níže na obrázku 2.6:

```
1: Inicializace populace vážek  $X_j(i = 1, 2, \dots, n)$ 
2: Inicializace krokového vektoru  $\Delta X_i(i = 1, 2, \dots, n)$ 
3: while není splněna podmínka do
4:   Spočítej kritériální hodnoty všech vážek
5:   Aktualizace zdroje potravy a nepřítele
6:   Aktualizace  $w, s, a, c, f, e$ 
7:   Spočítej S,A,C,F,E dle rovnic 2.13 až 2.17
8:   Aktualizace sousedství
9:   if má vážka alespoň jednoho souseda then
10:     Aktualizace vektoru rychlosti dle 2.18
11:     Aktualizace pozičního vektoru dle 2.19
12:   else
13:     Aktualizace pozičního vektoru dle 2.20
14:   end if
15:   Kontrola a úprava nové polohy na základě hranic proměnných
16: end while
```

Obr. 2.6: Pseudokód DA algoritmu [35]

2.2 Gradientní metody

2.2.1 Metoda nejstrmějšího sestupu

Metoda nejstrmějšího sestupu patří mezi nejjednodušší gradientní metody. K jejímu výpočtu potřebujeme znát analytické vyjádření účelové funkce, nebo alespoň odhad optima. Snažíme se najít jediné minimum funkce, které hledáme v opačném směru než je gradient účelové funkce, protože gradient je směr nejstrmějšího růstu. Jinak řečeno, jako směr hledání volíme zápornou hodnotu gradientu [14]

$$\overline{p_k} = -\overline{\nabla_k}. \quad (2.23)$$

V každé iteraci dochází k poklesu hodnoty minimalizované funkce. Nový odhad optima získáme z aktuálního odhadu tak, že se posuneme proti směru gradientu, jehož velikost násobíme délkou kroku α_k . Výsledný algoritmus [14] lze vyhádřit takto

$$\overline{x_{k+1}} = \overline{x_k} - \alpha \overline{\nabla_k}, \alpha_k \leq 1. \quad (2.24)$$

Volba optimální délky kroku není jednoduchá a je třeba ji zvolit tak, aby došlo k dostatečnému poklesu hodnoty kritériální funkce. Na začátku algoritmu volíme hodnotu α_k zpravidla větší a postupně ji snižujeme. Pokud je délka kroku hodně velká, mohlo by dojít k velkému poklesu hodnoty kritériální funkce a algoritmus by dlouhým krokem překročil minimum. Pokud by naopak byla délka kroku příliš krátká, tak bychom se k minimu blížili velmi pomalu a nehrozilo by, že bychom minimum přeskočili. Nevýhodou této metody je pomalá konvergence při volbě malého kroku α_k .

2.3 Metody vyšších řádů

2.3.1 Newtonova metoda

Newtonova metoda je gradientní metoda, která aproximuje optimalizovanou funkci v okolí výchozího bodu Taylorovým polynomem, který se na ni dívá jako na kvadratickou funkci. Jako nový odhad se používá minimum této paraboly (bod x^* , ve kterém pro funkci f platí $\nabla f(x^*) = 0$). Pro výpočet řešení v aktuálním kroku x_{k+1} budeme potřebovat aproximaci Hessovy matice $\nabla^2 f(x_k)$ v k -té iteraci, dále pak $\nabla^2 f(x_k)$ aproximaci první derivace v k -té iteraci a aproximaci řešení v předchozím kroku x_k . Celý postup je pak [13] vyjádřen vzorcem

$$x_{k+1} = x_k - (\nabla^2 f(x_k))^{-1} \nabla f(x_k). \quad (2.25)$$

Pokud je optimalizovaná funkce vyššího než druhého řádu, nemusí být nalezený bod přesně minimem funkce, ale předpokládá se, že se jedná o zlepšení oproti původnímu bodu. Tato metoda konverguje v případě, že je počáteční aproximace x_0 dostatečně blízko řešení x^* [15]. Ovšem nevýhoda této metody je, že nemusí konvergovat vůbec, a to v případě, že x_0 není dost blízko x^* . Dalším nedostatkem této metody je nutnost výpočtu inverzní Hessovy matice, což pro složitější úlohy může být výpočetně náročné a může se stát zdrojem dalších nepřesností. Nedostatky této metody řeší složitější algoritmy, například Kvazinevtonské metody.

3 Registrace obrazu u ultrazvukových dat

Ultrazvukové zobrazování (dále UZ) je neinvazivní metodou probíhající v reálném čase. Jde také o levnější alternativu k jiným lékařským zobrazovacím metodám a poskytuje mnoho výhod pro klinické i vědecké účely s aplikací jak na lidech, tak i na zvířatech. UZ pacienta nevystavuje ionizujícímu záření, je bezbolestný a poskytuje nepřetržité zobrazování videa v reálném čase. Využívá se v mnoha oblastech medicíny k zobrazování například lidského mozku, srdce či ledvin [18]. Ultrazvuková data získáváme buď to manuálním skenováním oblasti zájmu nebo podle konkrétního vzoru skenování prostřednictvím rotace a translace ultrazvukového měniče. Bez ohledu na vzor skenování ultrazvukového měniče je obvykle sonda přiložena k oblasti zájmu k získání co nejlepšího poměru signálu k šumu (SNR) pozorované tkáně. Ultrazvukové zobrazování vytváří nepřetržitý proud obrazů, kdy je často nutné tyto obrazy registrovat společně. Zvláště u klinického hodnocení celého orgánu. Velkou nevýhodou UZ je vysoký podíl šumu a obrazy tak podléhají několika unikátním zkreslením. Často využívaná je tedy multimodální registrace UZ společně s dalšími zobrazovacími metodami, jako jsou například MRI, CT či PET. Takovéto kombinace metod nám zvýší kvalitu zobrazování. Monomodální registrace ultrazvuku je náročná, ovšem omezuje náklady spojené s ostatními metodami, jako je například MRI, či potencionální rizika spojená s ionizujícími CT paprsky. Diagnóza za pomoci UZ může být například založena na hledání smysluplných rozdílů mezi ultrazvukovými snímky, které byly pořízeny měsíce nebo roky od sebe. Například populační studie zkoumají ultrazvukové obrazy mezi velkými počty subjektů k porovnání normální vs. patologické tkáně[18].

V registraci ultrazvukového obrazu se běžně vyskytují situace, u kterých se využívá rigidních transformací. První situací je registrace nedeformovatelných nebo neelastických anatomických struktur, jako jsou kosti. Druhou situací je prostá aproximace, využívaná pro hlubší orgány, například játra, kde nedochází k velkým změnám. Na druhé straně, častou volbou u registrace ultrazvukových sekvencí jsou elastické deformace, kde se využívají B-splajny [24]. Volí se právě kvůli deformovatelné povaze měkkých tkání, jako jsou cévy, orgány či mozek.

3.1 Odstraňování šumu

Jedním z nejběžnějších typů šumu v ultrazvukových obrazech je tzv. speckle, který je způsoben interferencí koherentní ultrazvukové vlny rozptýlené ve tkáni. Pokud registrace není založena na tomto typu šumu [19], tak nám speckle zhoršuje kvalitu UZ a bez jeho potlačení tak může docházet ke špatné registraci. Pro odstranění se využívá Gaussovske filtrace. Aby při odstraňování šumu nedocházelo k odstraňování

důležitých vlastností, jako například hran, tak se používá banky filtrů [20],[21]. Tato banka obsahuje Gaussův filtr, filtr první derivace (gradientní operátor), filtr druhé derivace, Laplacián a operátory s prahovými hodnotami. Dalším často používaným filtrem, který potlačuje speckle je tzv. mediánový filtr. Filtr pracuje jako nelineární dolní propust, která každému pixelu přiřadí medián získaný z jeho sousedních hodnot.

3.2 Zvýšení přesnosti registrace UZ obrazů

Mnoho registračních algoritmů zahrnuje ve svém postupu segmentaci obrazu a metody extrakce příznaků za účelem vedení registrace. Za pomoci segmentace jsou v ultrazvukovém obraze identifikovány prioritní oblasti zájmu, v jejichž okolí je prováděna registrace přednostně, zatím co regiony „pozadí“ jsou ignorovány, nebo registrovány poněkud omezeným způsobem. Automatická detekce příznaků využívá body zájmu v obraze společně s jejich lokálním sousedstvím, jako klíčové body pro registraci. Jedním z příkladů registrace založené na segmentaci je metoda[22]. Model byl ověřen na rozsáhlé databázi ultrazvukových snímků pořízených při snímání srdečního cyklu pacientů. Pomocí segmentace byla vytvořena 3D síť, kde se modul přizpůsobuje různým objemům a poskytuje přesnou bodovou korespondenci, která může být použita k odhadu tenkého splajnu pro výpočet transformace. Rychlost registrace tu závisí na rozlišení sítě a zůstává konstantní s rostoucí velikostí. Metoda není výpočetně náročná a velkou výhodou je její nezávislost na velikosti dat a nezáleží ani na intenzitě voxelů v obraze.

Zobrazení deformace myokardu umožňuje studovat jeho mechaniku a hodnotit řadu klinických scénářů, jako je například detekce ischemie myokardu či měření lékem indukovanou kardiotoxicitu spojenou s chemoterapií rakoviny. S klinickými scannery je nyní možné získat objemové datové sady a komplexnější, přesnější a časově efektivnější hodnocení. Pro analýzu ultrazvukových obrazů srdce, byly vytvořeny elastické FFD modely [24] založené na B splajnech, kde je třeba ruční segmentace endokardia a epikardia na konci diastoly. V těchto modelech byla deformace myokardu parametrizována obdélníkovou mřížkou FFD v kartézském prostoru. FFD model ovšem nemusí postihnout skutečnou anatomii srdce. Tuto nevýhodu potlačuje anatomicky volný model deformace AFFD, kde je B splajnová mřížka lokálně orientována podél směru povrchu srdce a je specifická pro každého pacienta. Vzhledem k účinnému pokrytí myokardu transformace vyžaduje méně parametrů a potenciálně klesne i registrační čas. V [24] studii byly modely aplikovány jak na syntetické 3D UZ data myokardu, tak na datech získaných od zdravých pacientů či pacientů po infarktu. Jiný, semi-automatický iterativní algoritmus segmentace byl uveden v [23], kde si uživatel ručně vybere počáteční body segmentace, aby získal strukturní

obrysy konkrétních objektů na obrázcích. Kriteriační funkce tohoto algoritmu zahrnuje velikost a směr gradientu, Cannyho detektor hran a Laplacián. Funkčnost algoritmu byla ověřena na synteticky deformovaných ultrazvukových snímcích z klinické databáze.

Pro zvýšení přesnosti UZ registrace může být také využito extrakce příznaků. V ultrazvukových obrazech se extrahují hrany a textury, což vede k plynulosti metrických funkcí, zlepšení optimalizace a zisku lepšího výsledku [25]. UZ registrační metoda, která je založena na porovnávání hranových map generovaných hranovým detektorem, byla popsána v [19], kde je detektor založen na teoretické distribuci plně formovaného specklu v UZ obraze. Registrace funguje jak pro UZ obrazy s nízkým kontrastem, tak i pro ty postižené specklem. Úspěšnost této registrace je vyšší, než u metod založených na gradientu. Speckle vzniká interferencí rozptylu UZ paprsků v buňce. Jeví se jako prostorově korelovaný šum, čehož se využívá u algoritmů sledující speckle. Metoda využívající speckle byla popsána u sledování pohybu myokardu in vivo ovci [19], kde se hlavním omezením ukázalo být přesný výběr oblasti zájmu pro registraci.

3.3 Perfúzní zobrazování

Perfuzi můžeme definovat jako průtok krve tkání či orgány. Zásobuje je kyslíkem a živinami, odvádí zplodiny metabolismu a je tak nezbytná pro správnou funkci celého organismu. Perfuzi lze jiným způsobem definovat jako rozdíl mezi arteriálním přítokem a arteriálním odtokem z uvažovaného objemu [26], tj. kapilární průtok.

Zobrazení a kvantifikace perfuze jsou důležité u pozorování fyziologických i patologických změn tkáně. Nejčastěji se používají pro diagnostiku nádorových onemocnění. Laserový Doppler, jednofotonová počítačová tomografie (SPECT), magnetická rezonance (MR) nebo multidetektorová počítačová tomografie (CT) [27]. Všechny tyto metody přesně kvantifikují tkáňovou perfuzi, ale jsou drahé a na víc je pacient vystavený záření nebo nukleárním látkám. Zobrazení perfuze pomocí ultrazvuku je díky jeho neinvazivnosti velice atraktivní a může být realizována dvěma způsoby. Prvním z nich je Dopplerovské zobrazení a druhým je využití kontrastních látek ve formě mikrobublin.

Dopplerovské zobrazení je jednoduché, levné a běžně využívané. Ovšem nelze jej využít k detekci kapilárního průtoku, který je velmi pomalý (méně než 1 mm/s). Vhodnou metodou pro kvantifikaci perfuze se ukázalo být využití mikrobublinových kontrastních látek. Mikrobubliny jsou charakteristické svou velikostí, která nepřesahuje 8 μm a umožňuje tak průchod bublinám do oblasti zájmu, dále pak prodlouženou perzistencí uvnitř kardiovaskulárního oběhu. Existuje řada klinicky schválených kontrastních látek různých složení. Obecně je mikrobublina ohraničena skořápkou,

která může být tuhá (např. denaturovaný albumin) nebo pružná (fosfolipidy) a má tloušťku 10 až 200 nm. Mikrobubliny jsou plněny plynem s vysokou molekulovou hmotností a nízkou rozpustností v krevním řečišti. Konkrétním příkladem je bublina vzduchu v lipidové či galaktózové obálce [28]. Mikrobublinový plyn je přibližně po 15 minutách po aplikaci injekce vydechován plícemi, zatímco složky obálky jsou metabolizovány játry nebo filtrovány ledvinami.

CEUS (Contrast-enhanced ultrasound) [27], je běžně v lékařské praxi používané kontrastní ultrazvukové vyšetření. Intravenózně aplikované plynové bubliny se dostanou do systémového oběhu. Mikrobubliny mají oproti tkáním vysoký stupeň echogenicity. Krev je původně hypoechogenní, po aplikaci kontrastní látky se mikrobubliny stanou odražeči a tím se zvýší modulace v prokrvených oblastech organismu. Během šíření ultrazvukové vlny, dochází ke kmitání molekul s kontrastní látkou. V místě polohy částice vznikají malé tlakové změny, které nazýváme akustickým tlakem. Při hodnotách tlaku menších než 50 kPa dochází k rytmické oscilaci mikrobublin na rezonanční frekvenci a jejich odpověď je maximální. Mezi 50 a 500 kPa vzniká nelineární odezva, která zapříčiní asymetrické změny ve velikosti mikrobublin a vytváří harmonické a subharmonické signální komponenty. Nakonec při akustických tlacích větších, než 500 kPa dochází ve většině případů ke zničení kontrastních bublin [28]. Mírou záporného akustického tlaku je charakterizován takzvaný mechanický index (MI), který obecně vyjadřuje stupeň nebezpečí poškození tkáně kavitací. S rostoucím MI se zvyšuje kvalita obrazu, ovšem při hodnotách vyšších 1.9 hrozí riziko prasknutí mikrobublin, které může mít za následek lokální prasknutí mikrovaskulatury a hemolýzu.

Kontrastních látek se využívá i u vysokoenergetického, dopplerovského zobrazení. Vysokoenergetický ultrazvukový pulz (s vysokým MI) vede k destrukci mikrobublin, při které je emitováno krátké silné snadno zaznamenatele zpětné echo. V současné době se již techniky, u kterých dochází k destrukci mikrobublin nepoužívají. Základní nosná ultrazvuková frekvence je eliminována a jsou detekovány nelineární frekvence vznikající odrazem od mikrobublin.

V rámci měření perfúzních dat jsou získávány sekvence ultrazvukových snímků, které mohou mít až stovky obrazů. Během snímání zpravidla dochází ke vzniku různých pohybů, které můžeme popsat dvěma základními mechanismy. První skupinu tvoří pohyby sondy v rukou lékaře během vyšetření samotného, díky kterému dochází ke změně zobrazovací roviny během akvizice. Druhou skupinu tvoří různé pohyby těla pacienta, popř. jednotlivých orgánů. Typickým příkladem jsou pohyby v důsledku dýchání, srdeční činnosti nebo peristaltiky střev. Z pohledu registrace takto naměřených sekvencí je dalším faktorem změna kontrastu v důsledku průchodu kontrastní látky jednotlivými částmi těla pacienta.

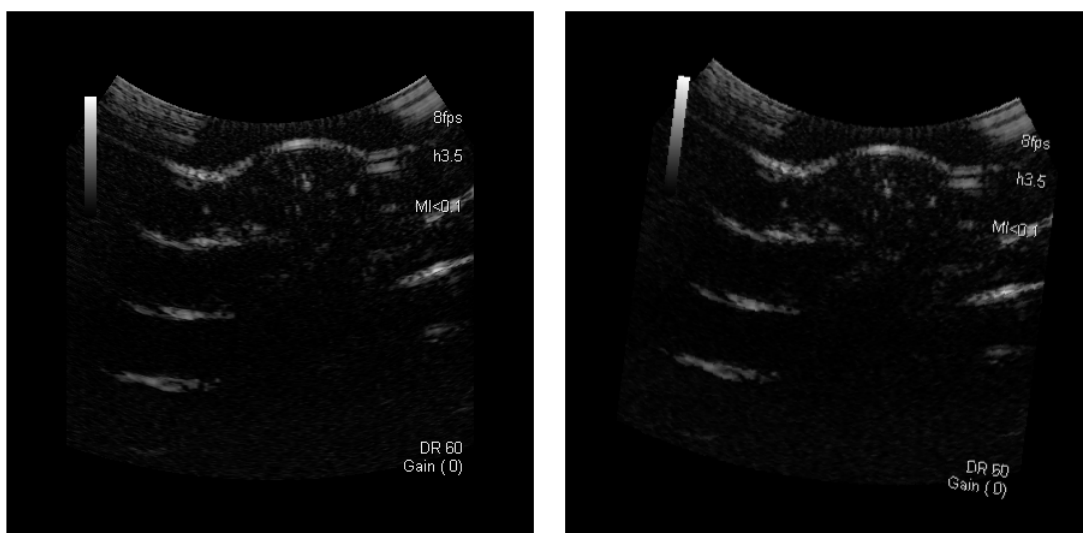
4 Realizace na simulovaných datech

Cílem této diplomové práce bylo implementovat některé optimalizační úlohy s vybranou kritériální funkcí v programovém prostředí Matlab a ověřit jejich funkčnost na testovací sekvenci. Tato kapitola obsahuje popis a srovnání pěti optimalizačních algoritmů, kterými jsou úplné prohledávání, Particle Swarm Optimalizace (PSO), včelí algoritmus (BA), algoritmus vážek (DA) a Nelder-Meadova heuristická metoda. Registrace probíhala na sadě šedesáti ultrazvukových snímků, kde byl uměle vytvořen náhodný pohyb v osách x, y a rotace. Registrace vždy probíhala mezi dvojicí obrazů referenčního a náhodně vychýleného. Cílem bylo pomocí implementovaných metod zajistit optimální registraci celé sady snímků.

4.1 Testovací sekvence

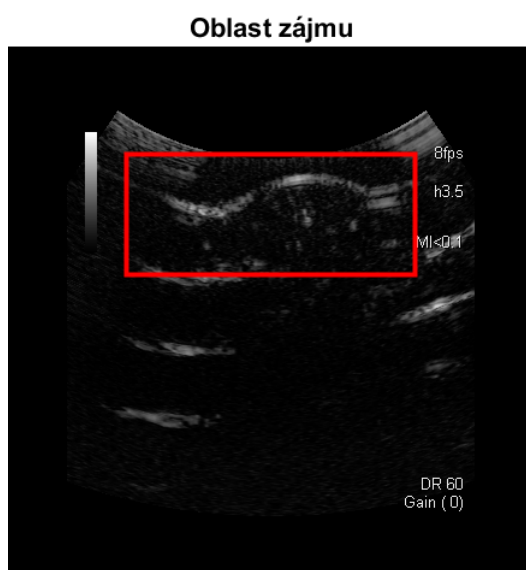
4.1.1 Předzpracování dat

Poskytnutá ultrazvuková data obsahovala 940 snímků. Z nich bylo vybráno 60 reprezentujících pro další postup registrace. Původní velikost obrázků byla 405 x 339 pixelů. U všech šedesáti byla oříznuta spodní část, která nesla nepodstatnou informaci pro proces registrace. Obrazy byly následně vloženy do matice nul, která omezí nežádoucí ztráty důležité informace. Do takto upravených snímků byl pomocí matlabovských funkcí uměle přidán posun v osách x a y v rozmezí 10 pixelů a rotace s úhlem natočení do 10° . Konkrétní vygenerované hodnoty posunu a natočení byly uloženy do pomocného vektoru pro výsledné zhodnocení procesu registrace. Na obrázku 4.1 je příklad referenčního obrázku a transformované scény.



Obr. 4.1: Ukázka referenčního obrazu (vlevo) a transformovaného obrazu (vpravo)

Samotný proces registrace probíhá v předem určené oblasti zájmu [3]. Toto obdélníkové okno bylo předem manuálně zvoleno pro danou sekvenci snímků. Ukázka zvoleného okna je na obrázku 4.2.



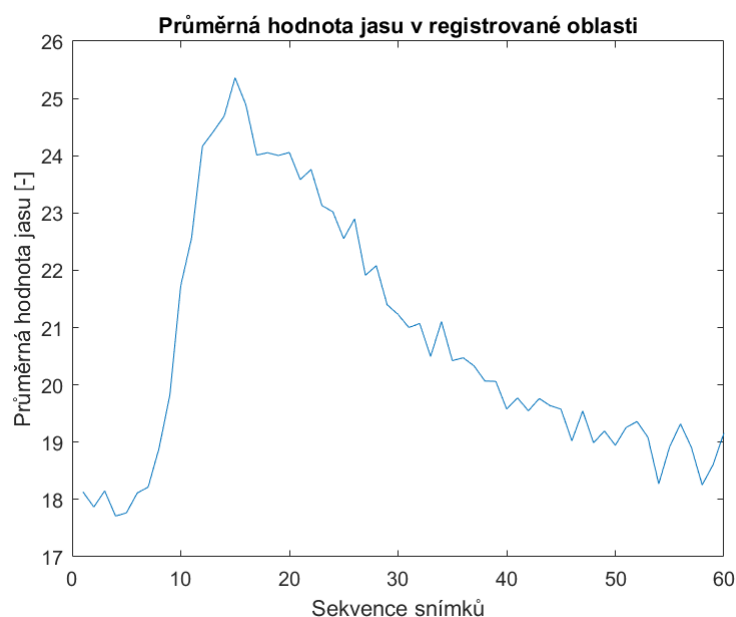
Obr. 4.2: Referenční snímek s obdélníkovým výřezem oblasti zájmu

4.1.2 Kriteriační funkce

Pro proces registrace je nezbytnou podmínkou vhodná volba kriteriační funkce. V kapitole 1.5 Metrika podobnosti jsou popsány kriteriační funkce používané u registrace ultrazvukových sekvencí. Pro vyjádření podobnosti obrazů byla zvolena metrika střední kvadratické odchylky (MSE) (výpočet dle vzorce 1.17), která je během optimalizačního procesu minimalizována.

4.1.3 Průběh kontrastu v sekvenci snímků

Pro nalezení optimálních parametrů transformace potřebných k registraci sekvence byla vybrána kriteriační funkce MSE, která je založena na hledání podobností mezi scénou a modelem v závislosti na intenzitě pixelů. Testovací sekvence byla pořízena po podání kontrastní látky, tudíž její koncentrace se v čase mění napříč sekvencí. Na obrázku 4.3 je zobrazena průměrná změna jasové hodnoty pixelů v obdélníkovém výřezu registrované oblasti.



Obr. 4.3: Graf znázorňující měnící se průměrnou hodnotu jasu v sekvenci snímků, v závislosti na množství kontrastní látky

Kontrastní látka v sekvenci nejprve roste, kde okolo patnáctého snímku dosahuje maxima a dále jen klesá. Taková změna koncentrace kontrastní látky má za následek změnu celkového jasu v obraze, což vede k ztížení dané optimalizační úlohy a nalezení správných parametrů geometrické transformace.

4.2 Metody optimalizace

Jádrem každé metody registrace obrazů je optimalizační proces, který zkoumá prostor geometrických transformací. Prohledávání se řídí metrikou podobnosti, což je kritériální funkce, která měří podobnost mezi zarovnávanými obrazy. Cílem optimalizačního procesu je nalezení nejlepších parametrů transformace, které povedou k správné registraci. Odhad transformace je řešen iteračním optimalizačním postupem, aby bylo možné správně prozkoumat hledaný prostor možných transformací. Každý optimalizační algoritmus má jinou strategii prohledávání. Aspekty jako například šum v obraze nebo diskretizace obazu způsobují potíže některým optimalizačním přístupům. Pro registraci obrazů se ukázaly být vhodnými metaheuristické algoritmy a algoritmy založené na evolučních výpočtech. Ačkoliv hrozí jejich uvíznutí v lokálních minimech a konvergence ke globálnímu optimu není jistá, jsou schopné provádět důkladné prohledávání ve složitých nedefinovaných problémech, jako je právě registrace obrazu. Ve své praktické části diplomové práce se věnuji metaheuristickým metodám rojení částic (PSO), včelímu algoritmu (BA) a algoritmu vážek (DA). Nejspolehlivější metodou pro získání globálního optima je úplné prohledávání přes celý obraz, ovšem za cenu vysoké výpočetní náročnosti. Pro srovnání byla tato metoda na testovací sekvenci také implementována.

4.2.1 Optimalizace metodou úplného prohledávání

Optimalizace metodou úplného prohledávání (z angličtiny exhaustive search) prochází veškeré hodnoty řešení a vybírá z nich to nejlepší. Ačkoliv je tato metoda jednoduchá na implementaci a pokud řešení existuje, vždy zaručuje jeho nalezení, tak má tato metoda i jistá omezení. Její výpočetní náročnost úměrně roste s počtem parametrů a velikostí prohledávaného prostoru. Čím hustší bude vzorkování prostoru, tím přesnější bude výsledná pozice optima, ovšem vyšší výpočetní čas.

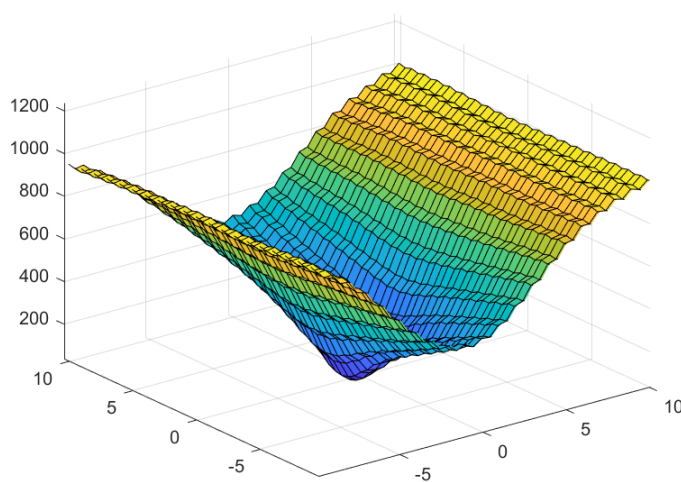
Vyhodnocení optimalizace úplným prohledáváním

Pro zjištění parametrů registrace bylo v této kapitole využito Brute Force algoritmu. Proces optimalizace proběhl pro registraci celé sady snímků. Registrace probíhala v obdélníkovém výřezu o velikosti 101x241 pixelů, který byl procházen s krokem 0,5. Jako kritériální funkci byla zvolena střední kvadratická odchylka (MSE). Její grafické zobrazení pro dvojrozměrný problém translace v osách x a y je znázorněn na obrázku 4.4. Celková úspěšnost algoritmu pro registraci jednotlivých dvojic obrazů byla vysoká. Pro srovnání hodnot simulované translace v osách x , y a rotace s hodnotami získanými optimalizačním procesem registrace, bylo využito výpočtu korelačního koeficientu. Jeho hodnoty jsou uvedeny v tabulce 4.1.

Tab. 4.1: Výstupy úplného prohledávání

	posun v ose x	posun v ose y	rotace
korelační koeficient	0,9988	0,9773	0,9997

Průměrná chyba registrace metodou úplného prohledávání byla pro posun v ose x 0,256 pixelů, pro hodnotu posunu v ose y 0,165 pixelů a pro rotaci 0,120 stupňů. Takto malé hodnoty jsou pro proces registrace zanedbatelné.



Obr. 4.4: Grafické znázornění kriteriální funkce MSE pro posun v osách x a y

Výpočetní náročnost algoritmu úplného prohledávání

Optimalizační proces úplného prohledávání byl výpočetně velice náročný. Výpočetní čas algoritmu pro celou sekvenci šedesáti snímků, pro výpočet pouze hodnot translace v osách x a y byl 225 sekund. Celkový čas pro zjištění parametrů translace i rotace pak vzrostl na 12727 sekund, což je v přepočtu přibližně 3,5 hodiny.

4.2.2 Optimalizace metodou PSO

Optimalizace rojem částic začíná s náhodnou populací jedinců. Částice v populaci iterativně mění své umístění v prostoru, který je prohledáván. Blíže je tento algoritmus již popsán v kapitole 2.1.

Výběr optimálních parametrů algoritmu PSO

Nalezení vhodných parametrů algoritmu je z hlediska efektivity registrace obrazů klíčové. Algoritmus tzv. roje částic má celkem 5 parametrů, které společně ovlivňují nalezený výsledek. Nejvýznamějšími z nich jsou počet jedinců a maximální počet iterací, které ovlivňují kvalitu řešení a výpočetní čas registrace. Dále jsou to váhy c_1 , c_2 a W , které slouží k úpravě pohybu jedinců v prostoru. Pro nalezení těchto parametrů bylo využito pěti dvojic obrazů z uměle zašuměné sekvence šedesáti snímků. Jednotlivé snímky se od sebe liší nejen translací v osách x, y a rotací, ale i hodnotami jasu, které se v sekvenci mění v závislosti na množství zachycené kontrastní látky. Na začátku sekvence jsou snímky bez kontrastu. Nejvíce kontrastní látky se nachází přibližně na dvanáctém snímku, v dalších snímcích množství kontrastní látky pouze klesá. Pro zjištění optimálních parametrů algoritmu byly využity dvojice obrazů zapsané v tabulce 4.3. Dle komparativní studie registračních algoritmů inspirovaných přírodou [34], byla určena rozmezí hodnot jednotlivých parametrů, uvedených v tabulce 4.2, se kterými probíhal proces optimalizace. Prostor pro hledání parametrů byl omezen pouze při generaci počáteční populace, a to na plus minus deset pixelů a úhlových stupňů.

Tab. 4.2: Parametry nastavení PSO

Parametr	Hodnoty
Počet jedinců	20, 30, 50, 100
Maximální počet iterací	10, 30, 50, 100, 200, 500
C_1	1, 2, 3, 4, 5
C_2	1, 2, 3, 4, 5
W	0, 0.2, 0.4, 0.6, 0.8, 1

Po spuštění optimalizačního algoritmu pro zvolenou dvojici obrazů, bylo po kombinaci všech nastavení získáno celkem 3600 hodnot kritériální funkce. Cílem optimalizace je minimalizace zvoleného kritéria. Na základě tohoto faktu bylo z těchto hodnot vybráno 20 nejmenších. Průměrné hodnoty kritériální funkce ze všech dvaceti nastavení a jejich směrodatné odchylky jsou v tabulce 4.3. Výsledná hodnota směrodatné odchylky kritériální hodnoty v každé z pěti dvojic obrazů byla průměrně 0.0006 a hodnota kritériální funkce se pohybovala v rozmezí 48 - 173.

Tab. 4.3: Střední hodnota a směrodatná odchylka kritériální funkce, získané z dvaceti nejlepších výsledků registrace dvojic obrazů pro různá nastavení parametrů PSO

Dvojice registrovaných obrazů	Kritériální funkce	
	Průměrná hodnota	Směrodatná odchylka
1-2	48.4717	0.0002
1-13	173.32	0.0020
1-28	102.5013	0.0002
12-13	73.3911	0.0002
27-28	57.6478	0.0005

Hodnota kritériální funkce pro různá nastavení parametrů PSO je závislá také na koncentraci kontrastní látky v obraze, jak lze vidět v tabulce 4.3. Koncentrace kontrastní látky v celé sekvenci není konstantní, což zhoršuje přesnost registrace. Pokud na sebe například budeme lícovat 1. a 2. snímek, které neobsahují kontrast, dostaneme se při některých nastaveních PSO průměrně na hodnotu kritériální funkce 48,4717. U lícování 1. snímku bez kontrastu na 13., kde se pohybujeme v oblasti maximálního množství kontrastní látky, bude hodnota kritériální funkce o poznání vyšší - 173.32.

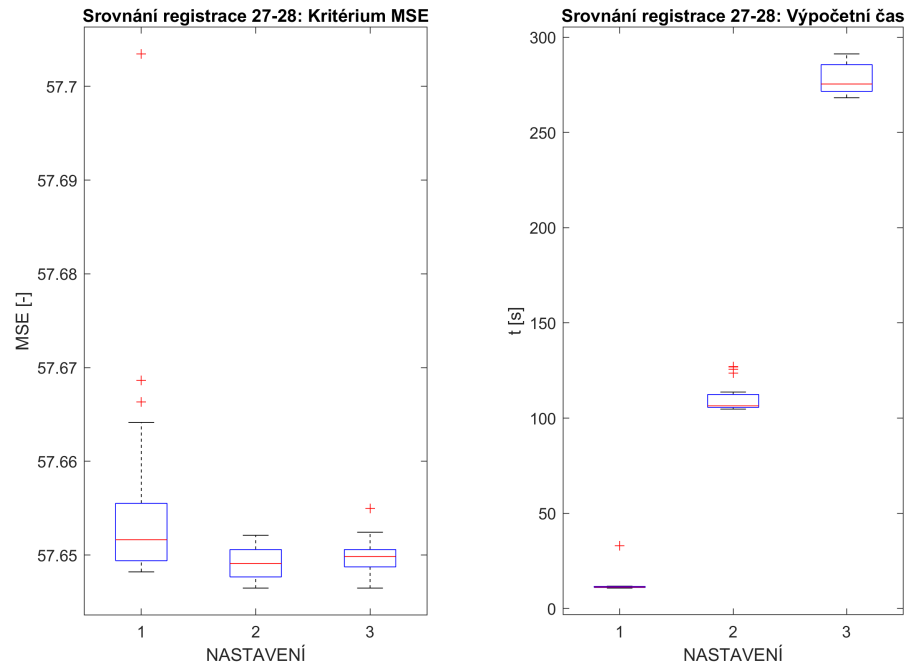
V tabulce 4.4 jsou zvolená tři nejčastěji zastoupená nastavení parametrů PSO. Byla vybrána na základě nízké směrodatné odchylky dvaceti hodnot kritériálních funkcí a zásadně se liší délkou výpočtu.

Tab. 4.4: Tři zvolená nastavení algoritmu PSO

Parametry algoritmu	Nastavení 1	Nastavení 2	Nastavení 3
Počet jedinců	20	100	100
Maximální počet iterací	100	200	500
C1	2	2	2
C2	1	1	2
W	0.6	0.6	0.6
Průměrná výpočetní náročnost [s]	11	115	286

Pro tyto tři konkrétní nastavení optimalizačního algoritmu byla opět spuštěna registrace všech pěti dvojic snímků. Jelikož se jedná o stochastický algoritmus, pracující s náhodně rozmístěnými jedinci v populaci, může se hodnota kritériální funkce po každém spuštění lišit. Algoritmus byl tedy spuštěn 30 krát s konkrétním nastavením, kde předmětem sledování byl rozptyl hodnot kritériální funkce. Na obrázku 4.5

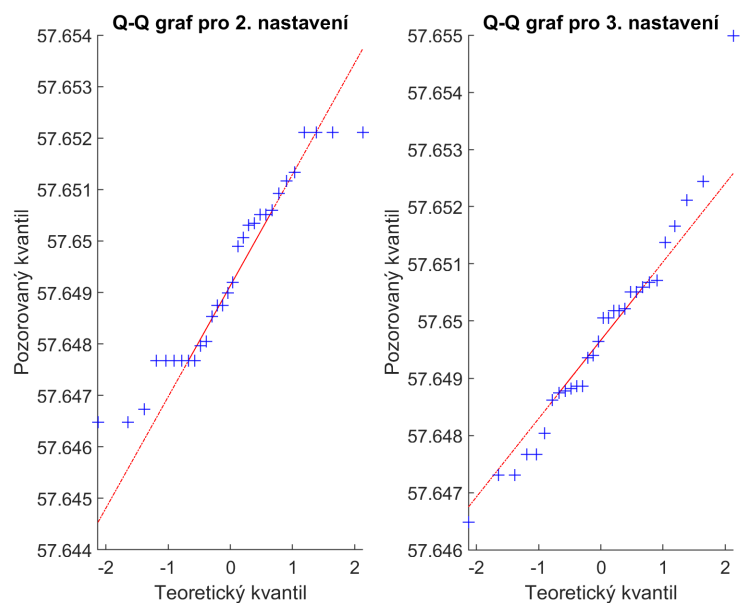
jsou boxplotem vyjádřeny výsledky optimalizace pro nastavení z tabulky 4.4 u registrace dvojice obrazů 27-28 z tabulky 4.3. Ostatní boxploty pro zbylé čtyři dvojice snímků jsou v příloze A.



Obr. 4.5: Registrace dvojice snímků 27-28, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost

Ikdyž bylo při všech třech nastaveních dosaženo přibližně stejné hodnoty kritériální funkce, počet jedinců a maximální počet iterací mají zásadní vliv na délku výpočetního času algoritmu. Při použití prvního nastavení algoritmu s dvaceti jedinci a sto iteracemi, byl získán výsledek za 11 sekund, ovšem s nejvyšším rozptylem hodnot. Přibližně stejných výsledků pak dosahují zbylá dvě nastavení, která se od sebe více jak dvojnásobně liší ve výpočetní náročnosti. Aby bylo možné z těchto dvou nastavení vybrat to s kratším výpočetním časem, byla vyslovena nulová hypotéza, že tyto dvě nastavení mohou být považována za rovnocenná.

Nejprve bylo pomocí Anderson-Darlingova testu zjištěno nenormální (negaussovské) rozdělení dat, získaných po registraci, pro všechny dvojice snímků. To, že zkoumaná data nepocházejí z normálního rozdělení je také znázorněno na obrázku 4.6 pomocí grafu založeného na porovnání kvantilů teoretického rozdělení a naměřených kvantilů - tzv. Q-Q grafu.

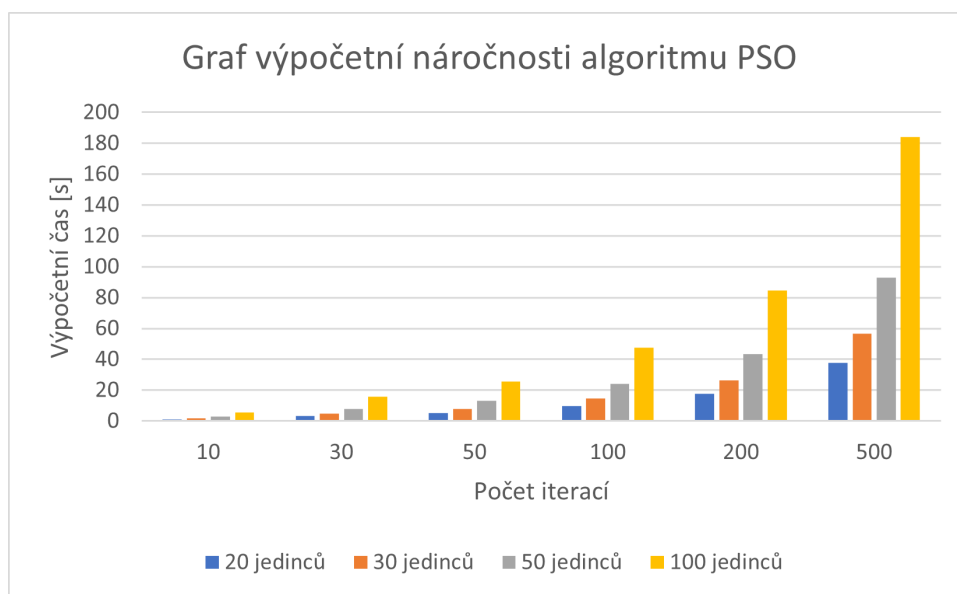


Obr. 4.6: Q-Q graf výstupu registrace dvojice snímků 27-28, vlevo 2. nastavení optimalizace, vpravo 3. nastavení optimalizace

Testování nulové hypotézy probíhalo neparametrickým dvouvýběrovým Kolmogorovo - Smithovým testem. Na hladině významnosti α rovné 0.05 vyšla pro každou dvojici registrace p hodnota v rozmezí 0,72-0,34. Nulová hypotéza tedy není zamítnuta, 2. a 3. nastavení algoritmu PSO lze považovat za rovnocenná. A proto kvůli kratšímu, potřebnému výpočetnímu času, bylo pro spuštění algoritmu na celé testovací sekvenci vybráno první a druhé nastavení.

Výpočetní náročnost algoritmu PSO

Cílem optimalizačního procesu je nalezení optimálního řešení za co nejkratší čas. U algoritmu PSO výpočetní náročnost přibližně lineárně roste společně s počtem jedinců a maximálním počtem iterací. V grafu na obrázku 4.7 je tato závislost zobrazena pro kritériální funkci MSE v případě registrace dvojic obrazů z tabulky 4.3.



Obr. 4.7: Graf časové náročnosti algoritmu PSO

4.2.3 Optimalizace rojem včel (BA)

Včelí algoritmus (BA) je jedním z meta-heuristických optimalizačních algoritmů. Je inspirován přirozeným chováním včel při hledání potravy. V zásadě včely provádí lokální prohledávání kombinované s globálním průzkumem.

Výběr optimálních parametrů algoritmu BA

Pro správné nastavení roje je třeba určit počet jedinců n (průzkumnic), počet z nich nejlepších včel m , počet elitních včel e , množství včel vyčkávajících, prohledávané okolí a maximální počet iterací. Dle doporučeného nastavení v článku zabývající se aplikací BA algoritmu [41] by počet elitních včel měl být menší, než je počet nejlepších průzkumnic a dále je doporučeno v okolí elitních jedinců zaměstnávat více průzkumnic, než kolem těch ostatních. Pro zjištění optimální kombinace parametrů pro simulovaná data byly zvoleny jednotlivé hodnoty algoritmu podle tabulky 4.5.

Tab. 4.5: Parametry nastavení BA

Parametr	Hodnoty
Počet včel (n)	20, 50, 100
Počet nejlepších včel (m)	20%, 30%, 50% z n
Počet elitních včel	25% , 50% z m
Počet vyčkávajících včel	5% ,10% ,20% z n
Velikost okolí	2, 5, 10
Maximální počet iterací	20, 50, 100, 200

Postup při hledání parametrů geometrické transformace s využitím roje včel probíhal stejně, jako u algoritmu PSO. Prostor hledání parametrů byl omezen pouze při generaci počáteční populace, a to na plus minus sto pixelů a plus minus padesát úhlových stupňů. Po spuštění procesu optimalizace pro jednu ze zvolených dvojic obrazů, vzniklo po kombinaci všech parametrů celkem 648 hodnot kritériální funkce.

Tab. 4.6: Střední hodnota a směrodatná odchylka kritériální funkce, získané z dvaceti nejlepších výsledků registrace dvojic obrazů pro různá nastavení parametrů BA

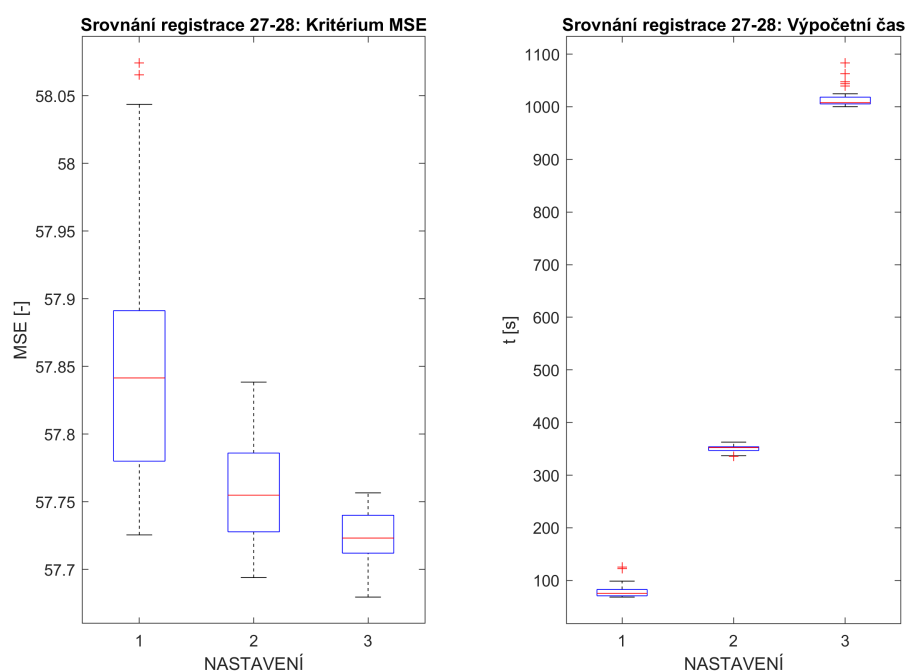
Dvojice registrovaných obrazů	Kritériální funkce	
	Průměrná hodnota	Směrodatná odchylka
1-2	48.5540	0.0121
1-13	48.5637	0.0139
1-28	48.5565	0.0118
12-13	73.4852	0.0165
27-28	57.7152	0.0138

Z pěti dvojic registrovaných obrazů, vyšla nejmenší průměrná hodnota kritériální funkce dle tabulky 4.6 pro dvojici snímků 1 a 2, které neobsahovaly kontrast. Nejvyšší hodnota kritéria pak vyšla při registraci dvojice obrazů 12 a 13, kde se naopak v obou snímcích nachází maximální hodnota koncentrace látky. Pro registraci dvojice obrazů 1 a 13, kde první snímek je bez kontrastu a druhý naopak s jeho maximální hodnotou, se minimální hodnota kritéria, spočítaného ze všech kombinací zvolených parametrů, rovnala hodnotě 48.5637. Tato hodnota je na setiny stejná, jako průměrná hodnota registrace dvojic obrazů neobsahujících kontrast. V tabulce 4.7 jsou vybraná tři nastavení algoritmu BA.

Tab. 4.7: Tři zvolená nastavení algoritmu BA

Parametry algoritmu	Nastavení 1	Nastavení 2	Nastavení 3
Počet včel	50	100	100
Počet nejlepších včel	25	30	50
Počet elitních včel	7	15	25
Počet vyčkávajících včel	10	20	20
Velikost okolí	1	1	1
Maximální počet iterací	50	100	200
Průměrná výpočetní náročnost [s]	80	350	1017

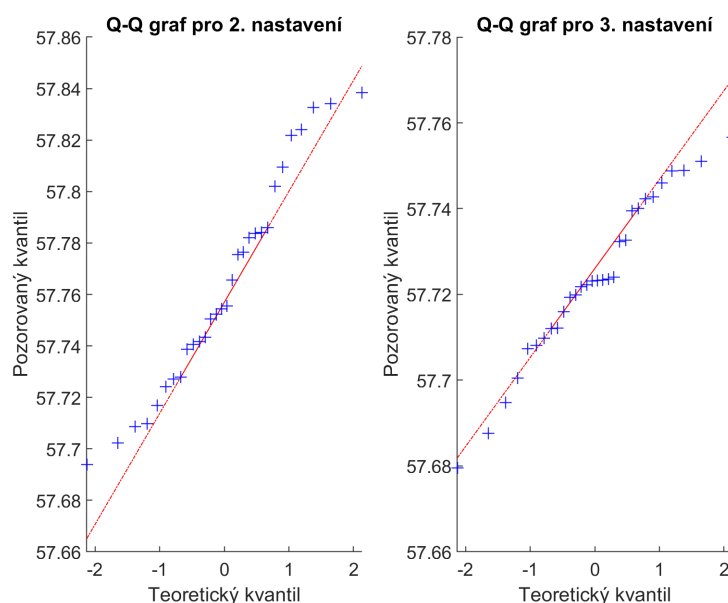
Algoritmus včelího roje byl pro každé nastavení z tabulky 4.7 třicetkrát spuštěn pro každou dvojici obrazů. Na obrázku 4.8 jsou boxplotem vyjádřeny výsledky registrace pro dvojici snímků 27-28. Boxploty pro zbylé čtyři dvojice jsou v příloze A.2.



Obr. 4.8: Registrace dvojice snímků 27-28, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost

Parametry modelu jsou vzorcem 4.1 shrnuty do jednoho čísla, kterým je vyjádřen počet včel v roji na iteraci. Pro první nastavení roj obsahoval 408 včel a nalezení

vhodných parametrů transformace trvalo přibližně 80 sekund oproti tomu 3. nastavení, které obsahovalo 1975 včel, počítalo přes 15 minut. Počet vyslaných včel tedy značně ovlivňuje výpočetní náročnost a pokud se najde vhodné rozdělení roje, pak lze dosáhnout na desetiny hodnoty stejné kritériální funkce za značně kratší výpočetní čas. Pomocí stejného postupu jako v kapitole 4.2.2 bylo zjištěno nenormální (negaussovské) rozložení dat, získaných opakovaným spuštěním registrace dvojice obrazů pro druhé a třetí nastavení včelího algoritmu. Ukázka nenormálního rozložení dat registrace dvojic obrazu 27-28 je znázorněna na obrázku 4.9 Q-Q grafem. Dle Kolmogorovo-Smithonova testu na hladině $\alpha = 0.05$ byla nulová hypotéza zamítnuta, tudíž druhé a třetí nastavení algoritmu BA nelze považovat za rovnocenná.



Obr. 4.9: Q-Q graf výstupu registrace dvojice snímků 27-28, vlevo 2. nastavení optimalizace, vpravo 3. nastavení optimalizace

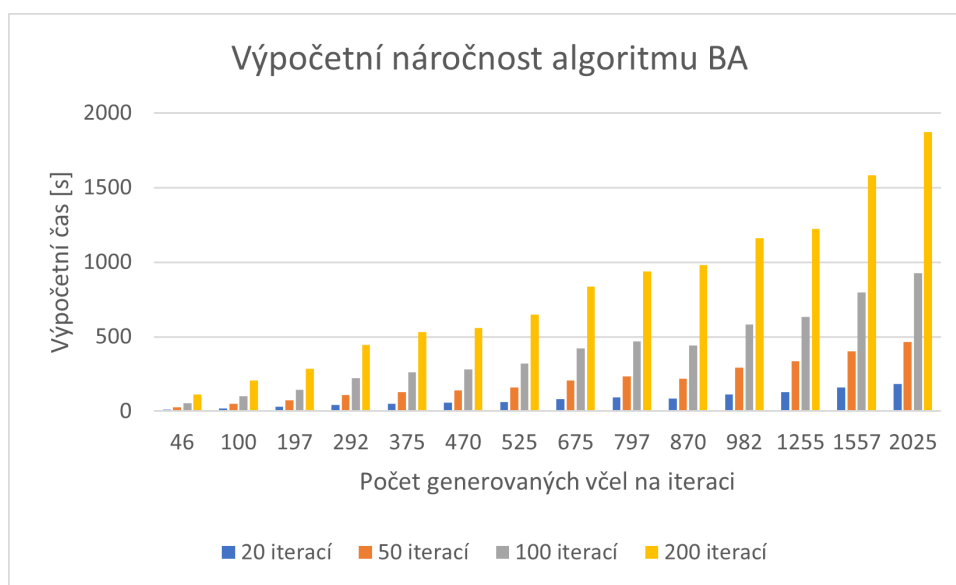
Výstupy jednotlivých nastavení metody BA se liší pouze v řádu desetin hodnoty kritériální funkce, což jsou v oblasti registrace pouze desetiny pixelu. Ale jejich výpočetní čas je zásadně rozdílný. Na základě těchto poznatků bylo pro registraci celé sekvence zvoleno 1. a 2. nastavení roje včel. Výstupem takto rozložených rojů je přibližně o 3 minuty kratší doba optimalizace, než by tomu bylo s 3. nastavením.

Výpočetní náročnost algoritmu BA

U optimalizace rojem včel mají jak na výsledek, tak i na délku výpočetního času vliv všechny nastavované parametry algoritmu uvedené v tabulce 4.5. Pro výpočet časové náročnosti metody byl tedy sestaven vzorec 4.1, který definuje celkový počet generovaných včel na iteraci (M).

$$M = e \cdot 2 \cdot vyck + m \cdot vyck + (n - m - e) \quad (4.1)$$

, kde n je celkový počet jedinců, e jsou vybrané elitní včely, $vyck$ je počet vyčkávajících a m je zvolený počet nejlepších včel. V grafu na obrázku 4.10 je patrná téměř lineární závislost mezi maximálním počtem iterací, počtem jedinců a výpočetní náročností algoritmu.



Obr. 4.10: Graf časové náročnosti algoritmu BA

4.2.4 Optimalizace algoritmem vážek (DA)

Dalším rojovým algoritmem, využívaným mimo jiné pro zpracovávání obrazů [39], je algoritmus inspirovaný chováním vážek v přírodě. Proces optimalizace je zahájen vytvořením sady náhodných řešení z ohraničeného intervalu hodnot. V každé iteraci se poloha a krok každé vážky aktualizuje dle rovnic 2.18, 2.19, 2.20. Na základě výpočtu Euklidovské vzdálenosti mezi vážkou a jejím okolím, probíhá aktualizace vektorů X a ΔX . Aktualizace pozice probíhá dokud není dosaženo maximálního počtu iterací.

Výběr optimálních parametrů algoritmu DA

U metody simulující chování roje vážek se vybíralo pouze mezi dvěma parametry, a to celkovým počtem vážek a maximální hodnotou iterací. Algoritmus byl implementován a nastaven dle Mirjalili [35] [42], dále pak upraven pro tuto konkrétní optimalizační úlohu. Zvolené rozsahy hodnot jsou vypsány v tabulce 4.8. Ostatní parametry popsané v kapitole 2.1.7 ovlivňující chování částic, jako jsou separace, zarovnání, koheze, atrakce a rozptýlení, jsou počítány adaptivně během optimalizace.

Tab. 4.8: Parametry nastavení DA

Parametr	Hodnoty
Počet jedinců	20, 30, 50, 100
Maximální počet iterací	10, 30, 50, 100, 200, 500

Postup úlohy probíhal stejně jako u algoritmu PSO a BA. Prostor pro hledání parametrů byl omezen plus minus 100 pixely u translace a 100 úhlovými stupni pro rotaci. Na víc je v algoritmu implementováno zrcadlení kolem této hranice. Po kombinaci parametrů počtu jedinců a iterací vzniklo pro registraci jedné dvojice obrazů 24 hodnot kritériální funkce. Výstupy byly seřazeny dle velikosti a z nich bylo zvoleno 15 nejlepších. Skrze všech pět registrací byly vybrány parametry, které vedly k minimalizaci kritéria. V tabulce 4.9 jsou uvedeny průměrné hodnoty výsledné kritériální funkce pro dvojice registrovaných snímků.

Tab. 4.9: Střední hodnota a směrodatná odchylka kritériální funkce, získané z patnácti nejlepších výsledků registrace dvojic obrazů pro různá nastavení parametrů DA

Dvojice registrovaných obrazů	Kritériální funkce	
	Průměrná hodnota	Směrodatná odchylka
1-2	48.5015	0.0241
1-13	173.3855	0.0588
1-28	102.5858	0.0978
12-13	74.5605	0.0625
27-28	57.6664	0.0137

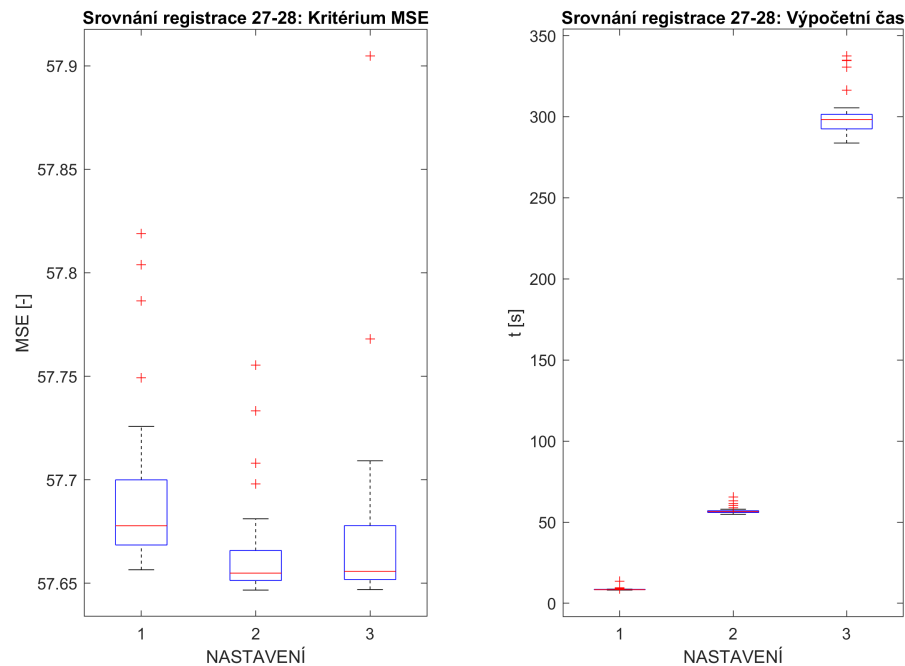
Stejně jako tomu bylo u předešlých metod optimalizace, jsou výsledné hodnoty kritéria získané metodou DA závislé na koncentraci kontrastu v jednotlivých snímcích. V tabulce 4.10 jsou vybraná tři nastavení, u kterých bylo dosaženo minimální

hodnoty kriteriální funkce, ale díky odlišnému počtu vážek a maximálnímu počtu iterací se značně liší délkou výpočetního času.

Tab. 4.10: Tři zvolená nastavení algoritmu DA

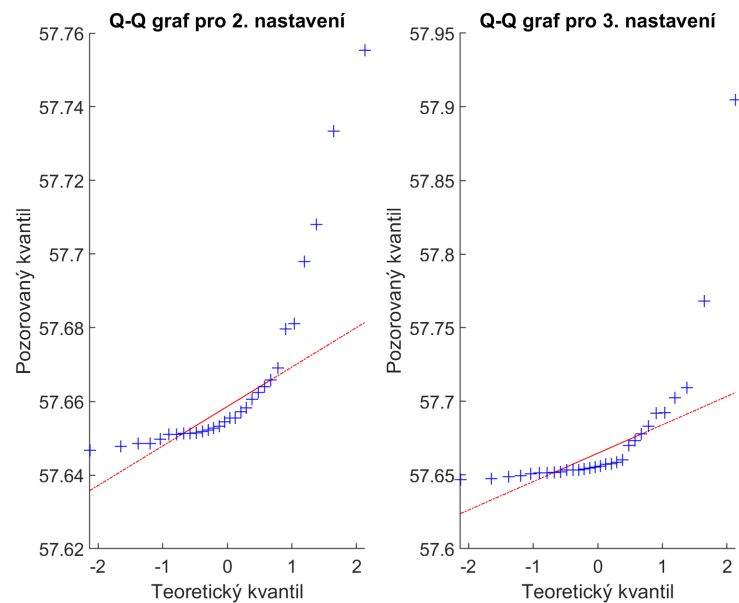
Parametry algoritmu	Nastavení 1	Nastavení 2	Nastavení 3
Počet jedinců	30	50	50
Maximální počet iterací	50	200	500
Průměrná výpočetní náročnost [s]	8	58	298

Vážky byly vyslány lovit kořist pro každé nastavení z tabulky 4.10 třicetkrát a takto pro každou dvojici snímků. Výsledný rozptyl hodnot kriteriální funkce pro konkrétní nastavení je na obrázku 4.11 vyjádřen boxplotem. Výsledné rozptyly hodnot kriteriálních funkcí pro zbylé dvojice snímků jsou v příloze A.3. U prvního a druhého nastavení se objevují odlehlé hodnoty MSE, které jsou způsobeny specifickou kombinací zvolených parametrů algoritmu, kdy se výsledek blíží nastavenému hraničnímu limitu a sklouzne do daného extrému.



Obr. 4.11: Registrace dvojice snímků 27-28, box plot vlevo: na ose x nastavení z 4.10, na ose y hodnota kriteriální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost

Na obrázku 4.12 je zobrazeno negaussové rozložení dat získaných po opakované registraci dané dvojice snímků. Použitím neparametrického Kolmorova-Smithonova testu byla zamítnuta hypotéza, která by statisticky potvrdila rovnocennost mezi 2. a 3. nastavením z tabulky 4.10.

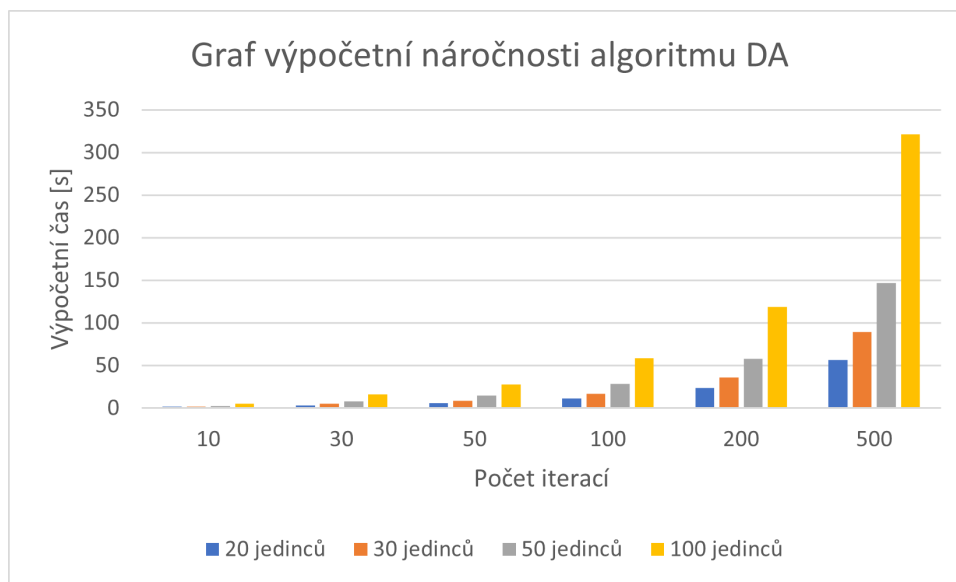


Obr. 4.12: Q-Q graf výstupu registrace dvojice snímků 27-28, vlevo 2. nastavení optimalizace, vpravo 3. nastavení optimalizace

Dále pro registraci simulované sekvence bylo vybráno první a druhé nastavení, která nejsou oproti třetímu nastavení tak výpočetně náročná.

Výpočetní náročnost algoritmu DA

Jelikož jsou v této metodě jednotlivé váhy pohybu vážky vypočítávány adaptivně, tak čas potřebný k zjištění parametrů transformace je ovlivněn pouze počtem částic v roji a maximální hodnotou iterací. Tato závislost je vyjádřena v grafu na obrázku 4.13, kde je mezi oběma parametry a časem patrná lineární závislost.



Obr. 4.13: Výpočetní náročnost DA

4.3 Vyhodnocení a srovnání algoritmů na testovací sekvenci

Cílem registrace je nalezení parametrů geometrické transformace, která optimálně připodobí dvojici obrazů ve smyslu minimalizace kritériální funkce střední kvadratické odchylky (MSE). Registrace probíhala na sekvenci šedesáti snímků, které byly uměle postihnuty náhodným posunem v osách x , y a rotací (viz. kapitola 4.1.1). Jako referenční obraz byl vždy brán první snímek ze sekvence, ke kterému se postupně transformovaly ostatní. Pro hledání optimálních parametrů transformační funkce byly použity již výše zmíněné metaheuristické algoritmy, a to roj částic (PSO), včelí algoritmus (BA) a algoritmus vážek (DA). Všechny zvolené přístupy se běžně využívají k řešení řady optimalizačních úloh a objevily se i v souvislosti s registrací medicínských obrazů [34].

Na kvalitu nalezeného řešení má vliv konkrétní nastavení daného algoritmu. Proto bylo provedeno hledání vhodných parametrů každého z nich. Nejdůležitějšími se globálně ukázaly být počet jedinců a maximální počet iterací. Společně s jejich

hodnotou rostla kvalita nalezeného řešení, ale zároveň i celková výpočetní náročnost. Na simulované sekvenci bylo provedeno testování dvou rychlejších nastavení, protože třetí nastavení na ukázkových dvojicích pro registraci nepředstavovalo další zlepšení a pouze prodlužovalo čas výpočtu. Všechna zvolená nastavení jsou popsána v kapitolách o nalezení vhodných parametrů u každé optimalizační metody. Jelikož byla data uměle zkreslena známými hodnotami translace i rotace, pro ověření správnosti registrace byl zvolen výpočet odchylky mezi skutečnými (a) a nalezenými parametry (b) transformace, dle vzorce

$$odchylka = \sqrt{(a - b)^2}. \quad (4.2)$$

V tabulkách 4.11 4.12 jsou zapsány průměrné hodnoty těchto tří parametrů, získaných na výstupu jednotlivých optimalizačních metod s použitými nastaveními pro všech 60 snímků sekvence.

Tab. 4.11: Srovnání skutečných hodnot parametrů geometrické transformace a parametrů získaných optimalizačními algoritmy s 1. nastavením

Algoritmus	Průměrná odchylka		
	Posun v ose x [px]	Posun v ose y [px]	Rotace [°]
DA	0.110	0.426	0.029
PSO	0.099	0.426	0.025
BA	0.097	0.421	0.030

Tab. 4.12: Srovnání skutečných hodnot parametrů geometrické transformace a parametrů získaných optimalizačními algoritmy s 2. nastavením

Algoritmus	Průměrná odchylka		
	Posun v ose x [px]	Posun v ose y [px]	Rotace [°]
DA	0.110	0.425	0.030
PSO	0.101	0.425	0.025
BA	0.099	0.422	0.027

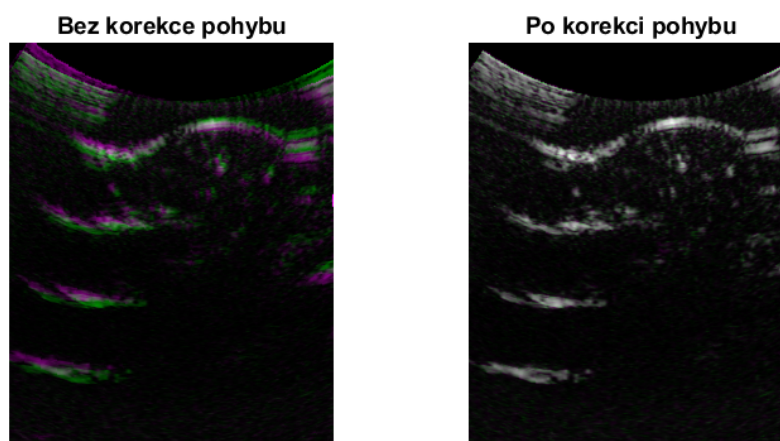
Nalezené parametry rigidní geometrické transformace obrazu se pro posun v osách x a y lišily nanejvýš o desetiny pixelu od původní hodnoty a úhel v řádu setin stupně. Další možností, jak srovnat kvalitu registrace, je výpočet střední kvadratické chyby (MSE) vždy mezi referenčním snímkem a na něj registrovaným, který je již posunut a natočen o nalezené parametry transformace. V tabulce 4.13 jsou zapsané

průměrné hodnoty MSE a celkového výpočetního času, který byl potřeba k nalezení parametrů geometrické transformace a registraci simulované sekvence.

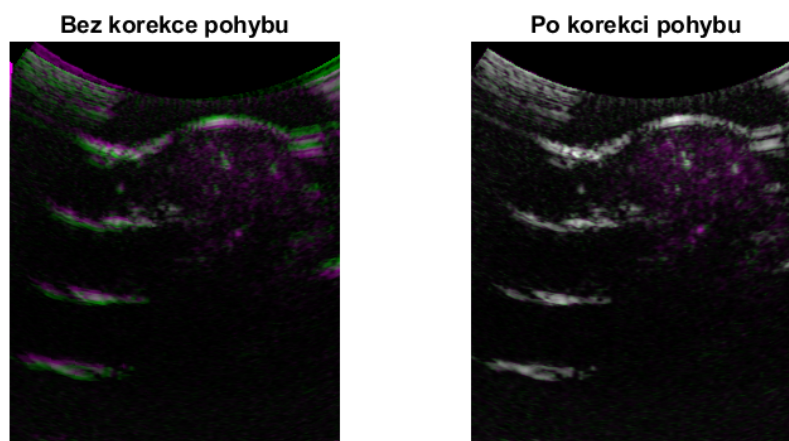
Tab. 4.13: Srovnání algoritmů

Algoritmus	Průměrná hodnota MSE		Celkový výpočetní čas [s]	
	1.nastavení	2.nastavení	1.nastavení	2.nastavení
DA	89.233	87.636	329	1971
PSO	89.099	87.596	416	3610
BA	89.349	87.733	4638	20494
Nelder-Mead	90.866	87.608	257	451

Na obrázku 4.14 je grafický výstup registrace dvojice snímku 1-2. Registraci sekvence ultrazvukových snímků ztěžovalo několik faktů, a to za prvé typický šum speckle vyskytující se pouze u ultrazvuku a za druhé koncentrace kontrastní látky, která se v průběhu sekvence mění. Nejvyšší hodnoty dosahuje okolo 12. snímku. Na obrázku 4.15 je tento rozdíl po registraci dvojice 1-12 viditelný.



Obr. 4.14: Registrace dvojice snímků 1-2



Obr. 4.15: Registrace dvojice snímků 1-12

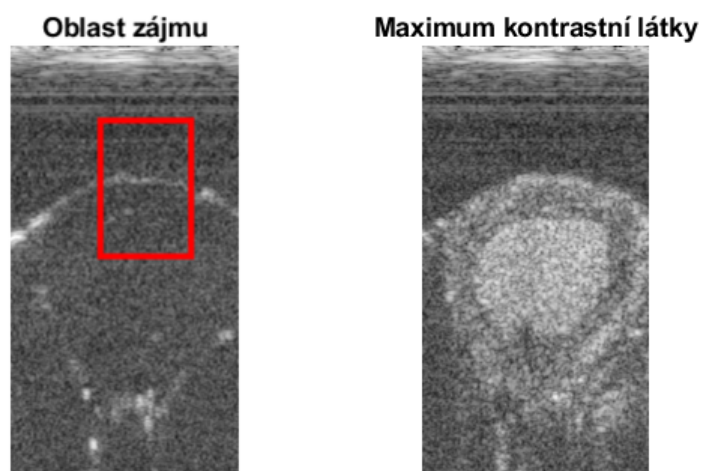
Spočítaná průměrná MSE původní sekvence, do které ještě nebyl uměle zanesen pohyb ani rotace, vyšla 88.333. Při registraci zašuměné sekvence bez jakékoli zpětné korekce pohybu, byla průměrná hodnota MSE 690. Po zjištění nejlepších parametrů transformace pomocí optimalizačních algoritmů a následné registraci, se hodnota MSE pohybovala okolo 89 pro 1. nastavení a 87 pro 2. nastavení (viz 4.13), kde se mezi sebou jednotlivé algoritmy lišily pouze v řádu desetin. Zásadní rozdíl těchto tří zvolených metod představuje výpočetní čas, a to v řádu desítek minut. Nejdéle trvala optimalizace roji včel, kde hledání parametrů a následná registrace zabrala při použití 2. nastavení přes 5 hodin, zatím co algoritmus s vážkami běžel kolem třiceti minut a s 1. nastavením dokonce necelých 6 minut.

Pro srovnání rojových algoritmů s ostatními, běžně využívanými technikami optimalizace, byl zvolen neevoluční Nelder-Meadův algoritmus, k jehož výpočtu byla využita v Matlabu implementovaná funkce *fminsearch*. V 1. nastavení, které je uvedené v tabulce 4.13, bylo k získání výsledku využito výpočtu pěti náhodných počátečních bodů, které byly generovány pomocí funkce *rand()*. Ve 2. nastavení byly body generovány systematicky, jako náhodné body v okolí vrcholů krychle, sestavené okolo počátku souřadnic všech registrovaných proměnných. Pro druhé nastavení bylo využito devíti bodů. Zvýšený počet odhadů vedl ke snížení kritéria na hodnotu 87.608, což je po PSO 2. nejlepší výsledek, získaný na testovací sekvenci. Zároveň tento heuristický algoritmus dospěl k optimálnímu výsledku za nejkratší čas.

Výsledné hodnoty střední kvadratické chyby prvního a druhého nastavení všech algoritmů se liší přibližně o 1.5, což vedlo k nalezení takových parametrů transformace, které se liší nanejvýš v řádech desetin pixelů či setin úhlu. Takto malé rozdíly hodnot posunu a natočení jsou pro proces registrace zanedbatelné. Co ovšem není zanedbatelné, je doba, za kterou byly tyto parametry zjištěny, a ta se mezi jednotlivými metodami mnohonásobně liší.

5 Preklinická sekvence

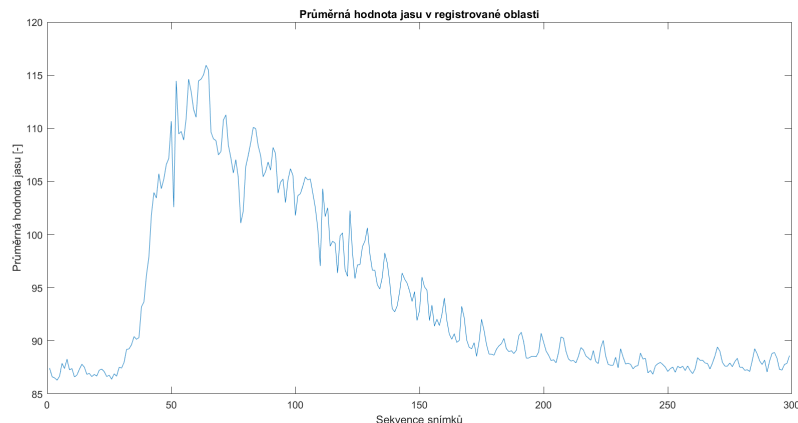
V této kapitole je popsán průběh registrace sekvence preklinických dat myokardu prasete. V rámci akvizice byla do těla prasete podána kontrastní látka SonoVue ve snížené dávce a proběhlo snímání ultrazvukových dat v krátké ose srdeční. Akvizice probíhala na otevřeném srdci se sondou přiloženou přes předsádku přímo na myokard. V následném zpracování dat byly pro další zpracování vybrány snímky, které odpovídaly R kmitu synchronně měřeného EKG. Výsledkem je dynamická sekvence zhruba 300 snímků. Na obrázku 5.1 je zvýrazněna obdélníková oblast, kde probíhala registrace. Velikost okna je 90 x 60 pixelů.



Obr. 5.1: Ukázka snímků ze sekvence

5.1 Průběh kontrastu v sekvenci

Jelikož se jedná o snímky pořízené po podání kontrastní látky, její koncentrace se v průběhu sekvence mění. Průměrné hodnoty jasu v obraze jsou znázorněny v grafu na obrázku 5.2, kde lze pozorovat nejjasnější pixely v rozmezí 50. až 80. obrazu v sekvenci. Dále průměrný jas klesá a na posledních snímcích se dostává zpět k hodnotám pod 90.



Obr. 5.2: Graf znázorňující měnící se průměrnou hodnotu jasu v sekvenci snímků, v závislosti na množství kontrastní látky

5.2 Vyhodnocení procesu optimalizace a registrace

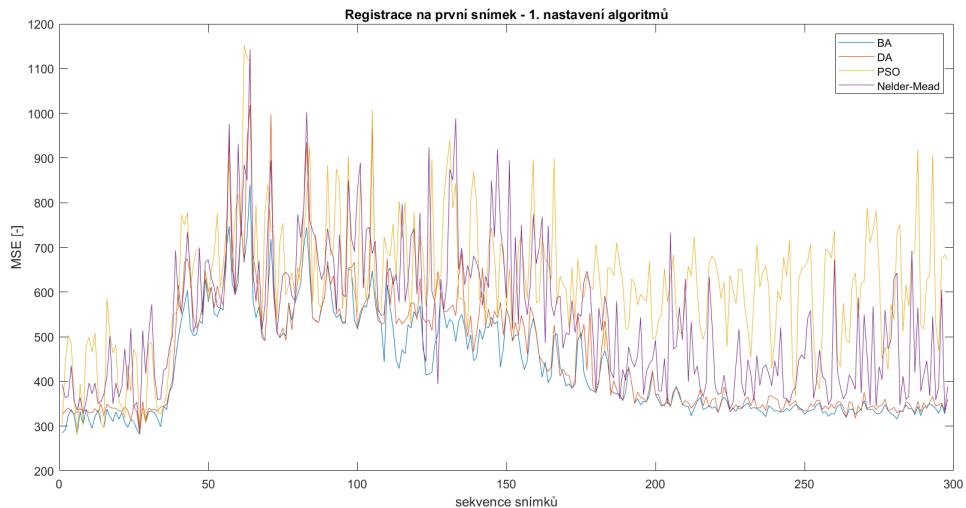
Cílem celé kapitoly 4 bylo nalezení vhodného nastavení optimalizačních algoritmů roje částic (PSO), včel (BA) a vážek (DA), které povedou k co nejlepší registraci. Dvě zvolená nastavení jednotlivých algoritmů byla použita pro celou preklinickou sekvenci. Postup registrace byl totožný s postupem aplikovaným na simulovaných datech, kde jako referenční snímek sloužil vždy první obraz ze sekvence, ke kterému se transformovaly ostatní. Ověřit správnost registrace u sady snímků, kde nejsou známy skutečné hodnoty vzniklé transformace, není tak jednoznačné, jako tomu bylo u simulované sekvence. Jako směrodatné kritérium je brána výsledná hodnota MSE na výstupu každého optimalizačního procesu. V souhrnné tabulce 5.1 jsou zaneseny průměrné hodnoty střední kvadratické chyby a také celková výpočetní náročnost všech algoritmů.

Tab. 5.1: Srovnání testovaných algoritmů na preklinických datech

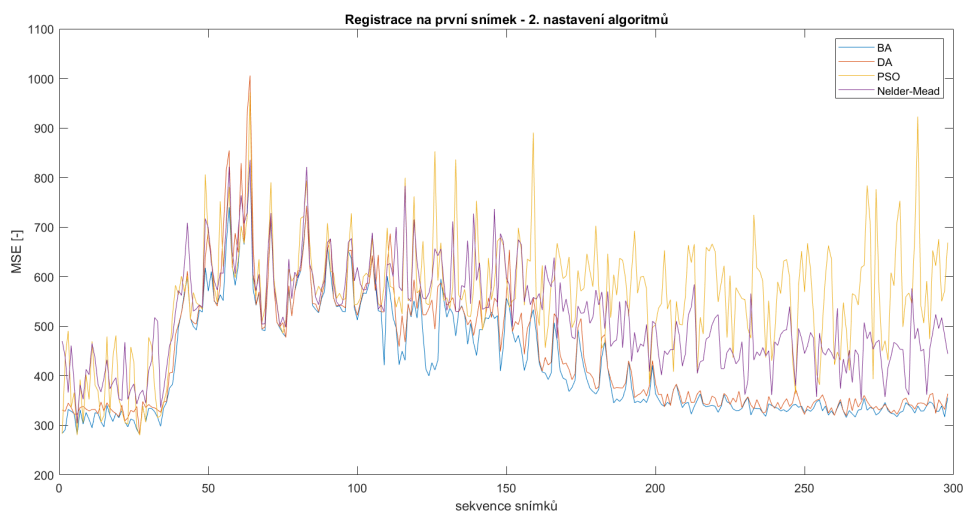
Algoritmus	Průměrná hodnota MSE		Celkový výpočetní čas [s]	
	1.nastavení	2.nastavení	1.nastavení	2.nastavení
DA	456.707	445.197	2178	14958
PSO	615.902	559.998	2815	27342
BA	430.530	424.334	30299	161353
Nelder-Mead	541.251	517.896	1106	1775

Při procesu optimalizace se podařilo hodnotu kritériální funkce mnohonásobně snížit. Průměrná hodnota MSE pro registraci celé sekvence bez korekce pohybu byla

1480. Dle hodnot v tabulce 5.1 lze globálně říct, že nejvíce se ji podařilo minimalizovat metodě roje včel (BA), která u 2. nastavení v průměru klesla až na 424.334. Nejhorší naopak dopadl algoritmus PSO, který vykazoval horší výsledky než referenční metoda Nelder-Meadova algoritmu. Algoritmus PSO vykazoval v porovnání s Nelderovým-Meadovým (NM) algoritmem výrazně vyšší časové nároky (až 90-násobný čas). Navíc průměrná výsledná hodnota MSE Nelder-Meada vyšla značně menší, než MSE roje částic. Průběh výsledné kritériální funkce v celé sekvenci není konstatní. Mění se v závislosti na koncentraci kontrastní látky v jednotlivých snímcích. Tato křivka je pro všechny optimalizační algoritmy graficky znázorněna na obrázcích 5.3 a 5.4. Pro algoritmy s prvním nastavením, se MSE podařilo minimalizovat až na hodnotu 280.366 a to s algoritmem PSO u registrace dvojice snímků 1-27. Zároveň nejhorší, z hlediska hodnoty kritériální funkce, dopadla optimalizace také metodou PSO, kde hodnota MSE u registrace dvojice 1-62 byla 1153, což je ve srovnání s globálně nejúspěšnější metodou BA rozdíl v hodnotě kritéria o 29, což znamenalo rozdíl hodnot translace v ose x 6, v ose y 4 pixely. Ovšem zjištěný rotační úhel se lišil o více než 30 stupňů. U algoritmů s 2. nastavením bylo globálního minima opět dosaženo algoritmem PSO, kde se jej podařilo pro stejnou dvojici snímků zmenšit o 2 desetiny. Naopak maximálního MSE 1006 bylo dosaženo rojením včel, u registrované dvojice 1-64, kde oproti globálně druhému nejúspěšnějšímu algoritmu DA, byla hodnota MSE, v případě 2. nastavení algoritmů, přibližně o 200 vyšší. BA se pravděpodobně nepodařilo uniknout z lokálního minima a hodnoty translace v ose x a úhlu, uvízly v nastavených hranicích algoritmu.



Obr. 5.3: Průběh MSE napříč sekvencí pro vybrané algoritmy s 1. nastavením



Obr. 5.4: Průběh MSE napříč sekvencí pro vybrané algoritmy s 2. nastavením

Největších rozdílů dosahovaly výsledky registrací mezi algoritmy PSO a BA, a to až ke konci sekvence (patrné v obrázcích 5.3 a 5.4 pro snímky 150 a více). Pro obě nastavení byl u dvojice snímků 1-289, zjištěn mezi kritériálními hodnotami průměrný rozdíl o 580. Výsledná transformace znamenala odlišné hodnoty translace v ose x a rotaci o jednotky pixelů a stupňových úhlů, v ose y dokonce o desítky pixelů. Ovšem tyto dvě metody nedávají na celé sekvenci zcela odlišné výsledky. K největší shodě došlo u registrace dvojice 1-15, kde rozdíl MSE s využitím 1. nastavení algoritmu, byl po zaokrouhlení pouhé dvě desetiny. A v tomto případě by PSO porazilo včely 10x rychlejším výpočtem. Pro přehlednost je vytvořena tabulka 5.2, která obsahuje srovnání parametrů, získaných jednotlivými metodami. Volba dvojice konkrétních algoritmů na vybraných dvojicích snímků, vychází z grafů na obrázcích 5.3 a 5.4.

Tab. 5.2: Tabulka výsledků registrací vybraných dvojic snímků ze sekvence

Algoritmy	Dvojice obrazů	Nastavení	Rozdíl na výstupu algoritmů			
			Posun x	Posun y	Rotace	MSE
BA-PSO	1-15	1	1.050	0.150	2.360	0.171
BA-DA	1-64	1	31.956	54.426	98.963	213.673
BA-DA	1-64	2	53.973	100.940	204.3807	472.187
BA-DA	1-106	1	47.109	56.510	101.150	328.651
BA-DA	1-106	2	1.882	0.015	1.371	4.297
PSO-NM	1-121	1	34.016	5.055	32.180	0.134
PSO-NM	1-215	2	25.071	30.005	5.366	0.332
BA-DA	1-265	1	0.047	0.064	0.336	0.042
PSO-NM	1-275	1	14.026	97.024	18.506	438.074
PSO-NM	1-289	2	13.984	31.979	35.610	426.520
BA-PSO	1-289	2	0.068	92.988	4.451	583.740

Z průměrné hodnoty MSE spočítané z celé sekvence a z grafů na obrázcích 5.3 a 5.4 plyne, že roj včel a vážek si z pohledu minimalizace kritériální funkce s optimalizací poradily nejlépe. Ne však u všech registrovaných dvojic konvergovaly do stejného optima. Například u registrace dvojice snímků 1-106, algoritmus vážek se svou hodnotou MSE téměř o 329 vyšší, než tomu bylo u BA, pravděpodobně divergoval. Rozdíl MSE u registrace této dvojice, stejnými algoritmy, pouze s využitím 2. nastavením, dopadla na jednotky rotace a posunu v ose x, nebo dokonce setiny posunu v ose y stejně.

To, že se hodnoty kritériálních funkcí, v průběhu sekvence mezi algoritmy někdy značně liší ještě neznámá, že daná metoda absolutně selhala. V takovýchto případech je třeba konkrétní dvojici obrazů najít a provést optimalizaci danou metodou opakovaně. Přeci jen, jedná se o algoritmy s náhodným rozmístěním počáteční populace, které jak je známo, mohou uvíznout v lokálním minimu. Tento postup by se zdál být vhodným pro registraci některých dvojic s použitím algoritmu PSO. Soudě dle grafů 5.3, 5.4, převážně pro dvojice od snímku 150 až do konce sekvence, kde se průměrná MSE pro tento algoritmus pohybovala okolo hodnoty 572 s parametry druhého nastavení. V těchto místech evoluční algoritmus dokonce předběhla Nelder-Meadova heuristická metoda, která v 2. nastavení, využívající 9 náhodných odhadů okolo vrcholů krychle, dosáhla průměrné MSE 483. I když se hodnotu kritéria v případě PSO nepodařilo u některých dvojic více minimalizovat, značný nádskok proti roji včel má v rychlosti výpočtu. Průměrný čas, potřebný k registraci dvojice obrazů algoritmem PSO se v případě 1. nastavení pohyboval okolo 9 sekund, zatímco u 2.

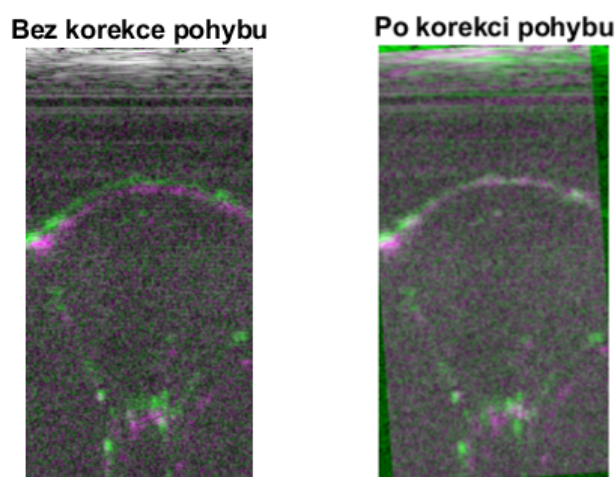
nastavení to bylo desetkrát déle (viz tabulka 5.3). I když metodou BA bylo kritérium globálně nejmenší, registrace pouhé dvojice snímků v průměru zabrala, v tom lepším případě, přes 100 sekund. Co se týče parametrů získané geometrické transformace, tak ty se v některých případech srovnání těchto dvou algoritmů, průměrně liší o jednotky pixelů u translace na ose x, desítky pixelů či úhlových stupňů u translace na ose y a rotaci.

Tab. 5.3: Průměrný čas potřebný k registraci jedné dvojice snímků ze sekvence

Algoritmus	Průměrný čas [s]	
	1. Nastavení	2. Nastavení
Nelder - Mead	4	6
DA	7	50
PSO	9	91
BA	101	540

5.2.1 Výstup registrace

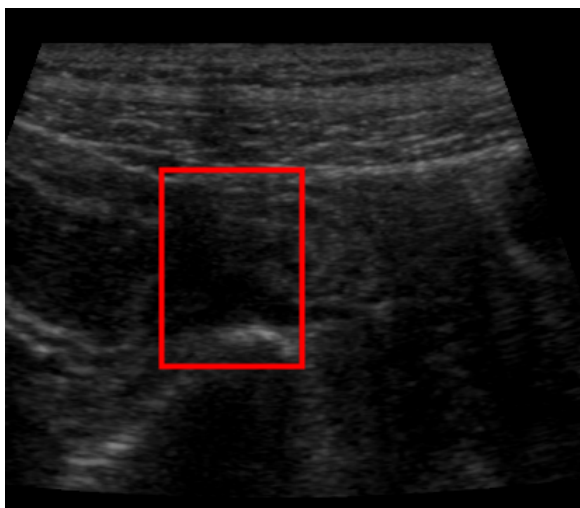
Z hlediska minimalizace kritériální funkce, všechny evoluční algoritmy dosáhly nejlepšího výsledku na stejné dvojici obrazů, a to 1-28. Hodnota MSE byla pro všechny 3 algoritmy těsně okolo 280 a nalezené parametry transformace se lišily nanejvýš v řádu desetin. Na obrázku 5.5 je zobrazen výsledek registrace před korekcí a po korekci pohybu, parametry nalezenými algoritmem BA.



Obr. 5.5: Registrovaná dvojice obrazů 1-28

6 Klinická sekvence

V případě klinické sekvence se jednalo o data tenkého střeva, která byla měřena pro hodnocení léčby pacientů s Crohnovou nemocí. Měření probíhala s podání kontrastní látky Sonovue. Registrace dat probíhala na datech z první harmonické komponenty, které nesou informaci o anatomických strukturách. Ve druhé harmonické jsou potom více vidět projevy kontrastní látky samotné. V rámci této práce byla sekvence upravena a vybráno pouze 100 snímků. Na obrázku 6.1 je vyznačena oblast zájmu.



Obr. 6.1: Ukázka snímku ze sekvence s vyznačenou oblastí zájmu

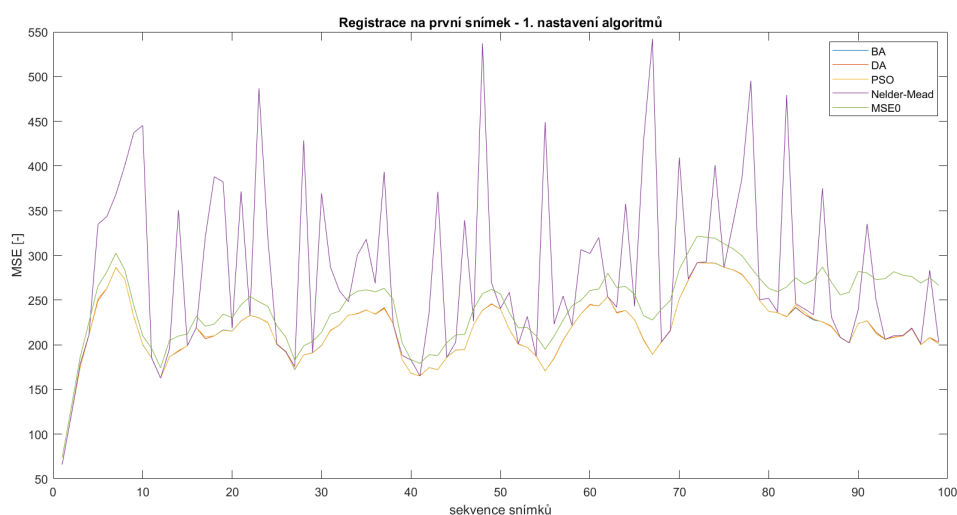
6.1 Vyhodnocení procesu optimalizace a registrace

Vybrané evoluční algoritmy byly s již zmiňovanými nastaveními (viz kapitola 4) spuštěny na klinických datech o délce 100 snímků. V průběhu registrace se opět pro každou dvojici obrazů hledaly, za pomoci dané optimalizační metody, nejlepší hodnoty posunu v osách x , y a rotaci, které vedly k zarovnání referenčního snímku se scénou. Z výsledků v tabulce 6.1 je patrné, že v průměru všechny evoluční algoritmy zareagovaly na sadu snímků na jednotky stejně.

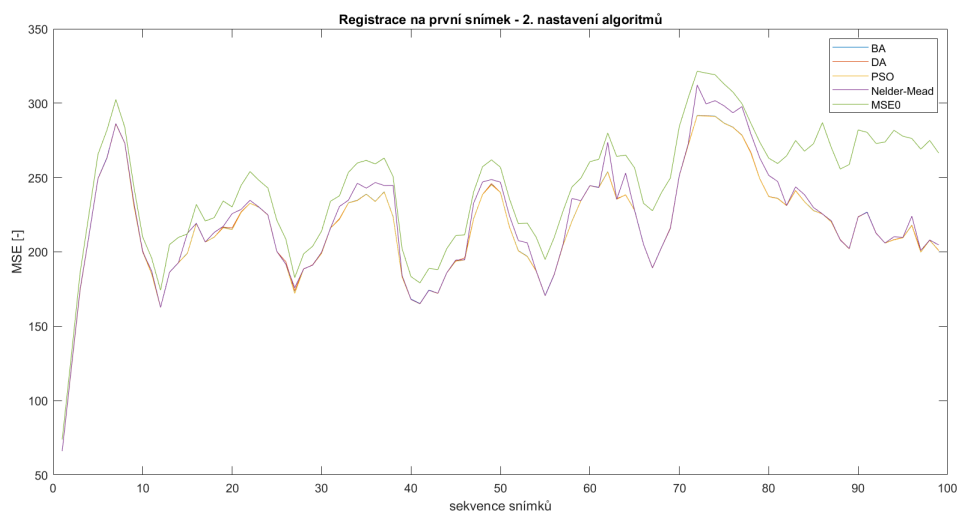
Tab. 6.1: Srovnání testovaných algoritmů na klinických datech

Algoritmus	Průměrná hodnota MSE		Celkový výpočetní čas [s]	
	1.nastavení	2.nastavení	1.nastavení	2.nastavení
DA	215.814	215.715	564	5790
PSO	215.759	215.553	673	8409
BA	215.760	215.668	7708	37493
Nelder-Mead	278.606	219.289	292	521

Jak je vidět v grafech na obrázcích 6.2, 6.3 , v procesu optimalizace se rojovým algoritmům podařilo původní hodnotu MSE minimalizovat. Bez korekce pohybu byla její průměrná hodnota 242. Po optimalizaci klesla na hodnotu 215. Průměrné hodnoty posunu a natočení nalezené procesem optimalizace, se pohybovaly okolo 3.5 pixelů v ose x, 0.4 pixelu v ose y a 0.9 úhlových stupňů.



Obr. 6.2: Průběh MSE napříč sekvencí pro vybrané algoritmy s 1. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu

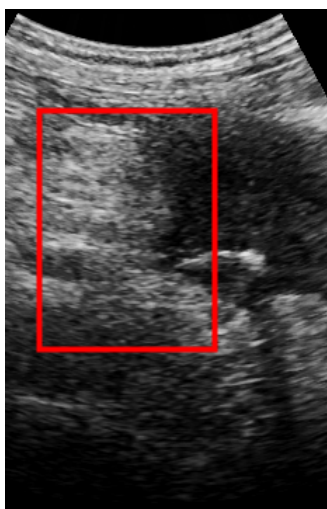


Obr. 6.3: Průběh MSE napříč sekvencí pro vybrané algoritmy s 2. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu

Pět počátečních náhodných odhadů, v případě prvního nastavení Nelder-Meadova algoritmu nestačilo ke konvergenci do lepšího řešení. V druhém případě, kde byly body systematicky generovány jako náhodné body v okolí vrcholů krychle, došlo k minimalizaci podobnostního kritéria. Ale oproti parametrům geometrické transformace nalezeným evolučními algoritmy, se jejich průměrné hodnoty zvýšily víc jak dvojnásobně. Zvláště pak rotační úhel, který vyšel v průměru pro celou sekvenci přes 5 stupňů. I když jsou výsledné MSE na výstupu všech evolučních algoritmů téměř stejné, co se liší, je jejich výpočetní náročnost. Zatímco včely (BA) hledaly optimální řešení pro registraci celé sekvence přes 10 hodin, nejrychlejší algoritmus vázek (DA) byl schopen dosáhnout na jednotky stejné hodnoty kritériální funkce za 90 minut.

7 Klinická sekvence 2

Jedná se o data nádoru pankreatu, u kterého byla dynamickou studií hodnocena míra prokrvení. Měření probíhala s podání kontrastní látky Sonovue. Registrace dat probíhala na datech z první harmonické komponenty, které nesou informaci o anatomických strukturách - především rozhraní slinivky a okolní měkké tkáně. Ve druhé harmonické jsou potom více vidět projevy kontrastní látky samotné. V rámci této práce byla sekvence upravena a vybráno pouze 100 snímků. Na obrázku 7.1 je jeden snímek ze sekvence zobrazen, spolu s vyznačenou oblastí zájmu.



Obr. 7.1: Ukázka snímku ze sekvence s vyznačenou oblastí zájmu

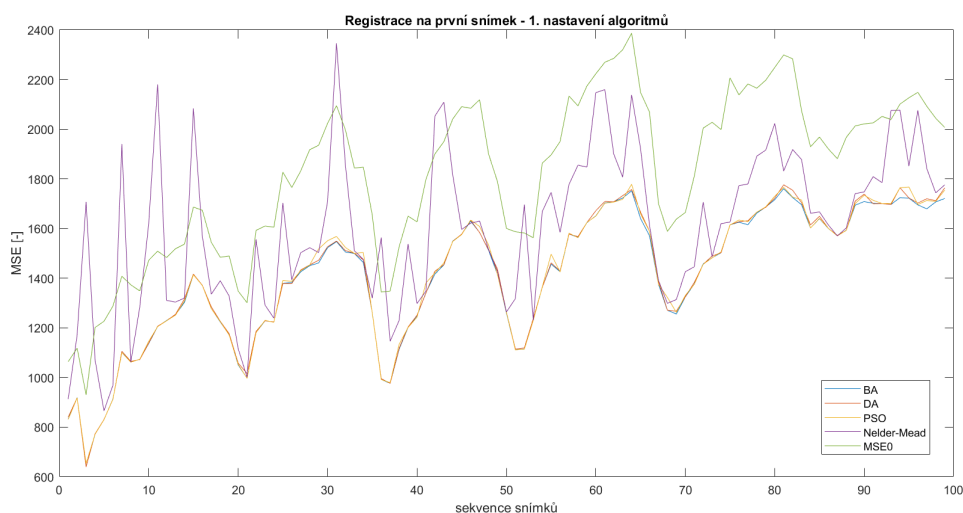
7.1 Vyhodnocení procesu optimalizace a registrace

Optimalizační algoritmy roje částic (PSO), včel (BA) a vážek (DA), byly dále spuštěny na dynamických datech nádoru pankreatu. Procesu optimalizace se opět podařilo minimalizovat podobnostní kritérium MSE z původní průměrné hodnoty 1843 na hodnoty zapsané v tabulce 7.1. Nejlepšího výsledku dosáhl roj včel (BA) druhého nastavení, ale jeho výpočet pro celou sekvenci trval více než 9 hodin, zatímco nejrychlejší evoluční algoritmus DA našel v průměru na desetiny hodnoty stejné parametry transformace do hodiny.

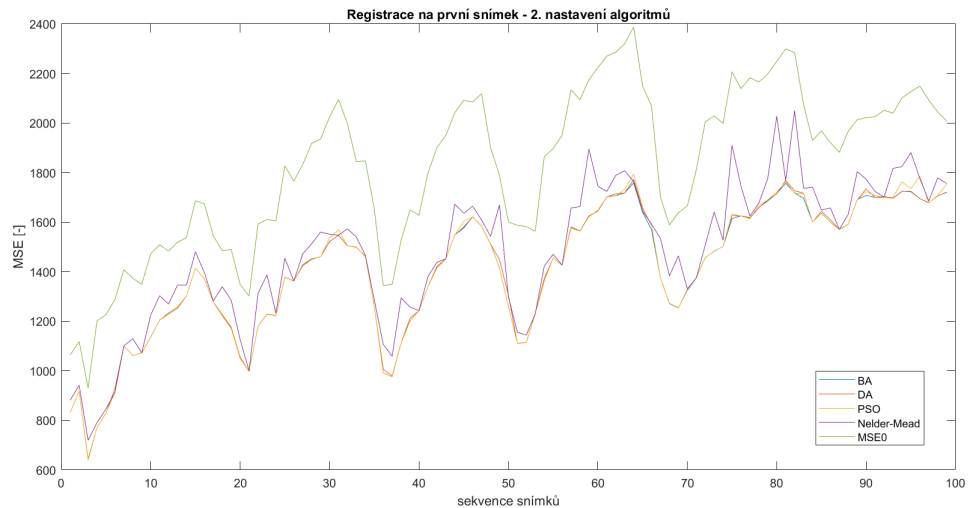
Tab. 7.1: Srovnání testovaných algoritmů na 2. klinických datech

Algoritmus	Průměrná hodnota MSE		Celkový výpočetní čas [s]	
	1.nastavení	2.nastavení	1.nastavení	2.nastavení
DA	1403.035	1399.137	422	3003
PSO	1404.610	1399.764	533	5297
BA	1397.096	1395.841	7034	33347
Nelder-Mead	1589.994	1463.367	258	441

Nalezené hodnoty parametrů se v průměru pro všechny evoluční algoritmy pohybovaly v rozmezí 0.3 - 38 px v ose x, 0.4 - 11 px v ose y a 0.5 až 11 úhlových stupňů pro rotaci. Na obrázcích 7.2 a 7.3 jsou graficky zobrazeny hodnoty MSE pro všechny registrované dvojice.



Obr. 7.2: Průběh MSE napříč sekvencí pro vybrané algoritmy s 1. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu

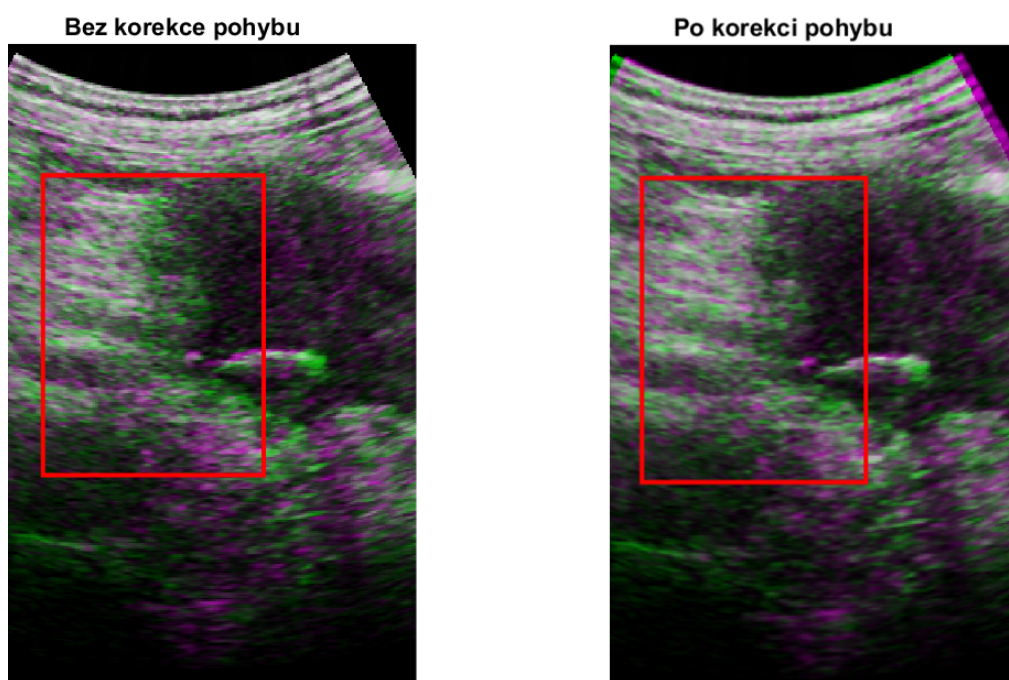


Obr. 7.3: Průběh MSE napříč sekvencí pro vybrané algoritmy s 2. nastavením, kde MSE0 je hodnota po registraci bez korekce pohybu

I když dle průměrné MSE z celé sekvence se jeví, že neevoluční algoritmus Nelder-Mead zvládl při optimalizaci aslepoň zmenšit hodnotu kritéria, tak dle křivky v grafu 7.2 je vidět, že v některých případech registrace 5 náhodných odhadů počátečního řešení k minimalizaci kritériální funkce vůbec nestačilo a algoritmus divergoval. V případě devíti náhodných odhadů v okolí vrcholů krychle, se Nelder-Meadovi v celé sekvenci podařilo minimalizovat MSE a přiblížit se tak výsledku získanému rojovými přístupy optimalizace, což je viditelné v grafu na obrázku 7.3.

7.1.1 Výstup registrace

Jako ukázka výsledku registrace po nalezení parametrů geometrické transformace, byla zvolena dvojice 1-38, kde se MSE na výstupu optimalizace v průměru pohybovala okolo 977 pro všechna nastavení. Na obrázku 7.4 je vlevo zobrazena registrovaná dvojice snímků před korekcí pohybu a vpravo registrace po korekci s nalezenými hodnotami transformace, algoritmem BA s druhým nastavením. Jako největší změnu našel algoritmus posun v ose x, a to o 8.5 pixelu. Je zde pozorovatelný rozdíl v oblasti, kde se registrace chytila na přechodu slinivky (černá oblast v obraze).



Obr. 7.4: Registrovaná dvojice obrazů 1-38

Závěr

Tato diplomová práce se věnuje registraci ultrazvukových snímků s využitím evolučních algoritmů. Postup registrace se skládá ze tří základních procesů, a to je nalezení vhodné transformace k propojení obrazů, výpočet metriky podobnosti, která popisuje kvalitu zarovnání a optimalizační proces, jež zajistí nejlepší nastavení zvolené transformace. Transformace scény, která vede k získání modelového obrazu, se dělí na dvě základní skupiny, a to na rigidní a elastické. Praktická část této diplomové práce se věnuje zpětnému zjištění parametrů rigidních transformací, kterými jsou translace v osách x, y a rotace.

V prvních třech kapitolách je teoreticky přiblížena daná problematika, jsou popsány různé optimalizační přístupy a práce s ultrazvukovými daty. Praktická část se věnuje registraci s využitím evolučních algoritmů roje částic (PSO), algoritmu včel (BA) a roje vážek (DA), kde podobnost dvou obrazů je vyjádřena kritériální funkcí střední kvadratické odchylky (MSE), která je v průběhu optimalizace minimalizována. V procesu registrace byly jednotlivé snímky vždy transformovány na první obraz ze sekvence. Jiné druhy registrace, jako například postupná transformace jednotlivých snímků na sebe, by do vyhodnocení mohly lehko vnést chybu, která by se například objevila při chybné registraci snímků hned na začátku sekvence a kumulovala by se až na výstup. Výsledek registrace závisí i na několika dalších věcech, jako je volba optimalizačního algoritmu, jeho konkrétní nastavení nebo velikost a umístění oblasti zájmu, ve které probíhá samotný proces hledání parametrů výsledné geometrické transformace. Registraci sekvencí ultrazvukových snímků navíc ztěžovalo několik faktů, a to za prvé šum speckle, vyskytující se pouze u ultrazvuku a za druhé obsah kontrastní látky, jejíž koncentrace se v průběhu sekvence mění.

V rámci řešení této diplomové práce bylo otestováno využití metaheuristických evolučních algoritmů pro registraci dynamických ultrazvukových sekvencí. Všechny tři navržené metody se ukázaly jako účinné pro využití na preklinických i klinických datech. V některých ohledech dosahovaly z pohledu zvoleného kritéria lepších výsledků ve srovnání s konvenčním postupem Nelderovým-Meadovým algoritmem ovšem za cenu vyššího výpočetního času.

Nejprve bylo v kapitole 4 provedeno hledání vhodných nastavení algoritmů, které vedlo k registraci simulované sekvence. Jelikož byly snímky uměle zkresleny známými hodnotami translace v osách x, y a rotací, pro ověření správnosti registrace byly tyto hodnoty porovnány s nalezenými hodnotami na výstupu optimalizace.

Zvolené algoritmy s konkrétními nastaveními byly spuštěny pro registraci preklinické sekvence snímků myokardu prasete, dále pak na dvou dynamických sekvencích vyšetření tenkého střeva a pankreatu. Jako směrodatné kritérium ověření správnosti registrace na klinických datech byla zvolena hodnota MSE, kterou se ve všech případech

padech optimalizace podařilo minimalizovat. Pro srovnání evolučních algoritmů s běžně využívanými metodami, byl použit heuristický Nelder - Meadův algoritmus z Optimization Toolboxu v Matlabu (funkce `fminsearch`), který se při vhodném rozložení počátečních náhodných odhadů, dokázal přiblížit výsledkům rojových algoritmů, za značně kratší výpočetní čas. Kde se algoritmy inspirované přírodou ukázaly být robustnější optoti Nelde-Meadovi, bylo při větších hodnotách posunů v jednotlivých osách. Již zmíněná výpočetní náročnost je dalším důležitým parametrem v procesu optimalizace. Ta hlavně závisí na počtu jedinců v roji a maximálním nastaveném počtu iterací daného algoritmu. Rozhodně časově nejnáročnějším se ukázal být algoritmus včel (BA), kde nalezení parametrů geometrické transformace pro klinické sekvence obsahující přibližně 100 snímků, trvalo téměř 10 hodin, zatímco nejrychlejší algoritmus vážek (DA), v průměru dosáhl téměř stejné hodnoty kritériální funkce za hodinu.

Hodnota MSE se napříč snímky v sekvenci mění. Její velikost je závislá na koncentraci kontrastní látky v obraze a účinnosti dané optimalizační metody v hledání parametrů geometrické transformace. Evoluční algoritmy se na některých dvojících snímků ukázaly být jako vhodná volba k nalezení parametrů posunů a rotace pro kvalitní výstup registrace, ovšem v některých místech sekvence nebyly dostatečně robustní. V takových případech by bylo na místě upravit nastavení daného optimalizačního přístupu pro konkrétní dvojice, popřípadě dodatečně opakovat optimalizaci úseku v sekvenci vícekrát.

Literatura

- [1] VALSECCHI, Andrea, S. DAMAS a Jose SANTAMARIA. Evolutionary Intensity-based Medical Image Registration: A Review: A Review. *Current Medical Imaging Reviews*. 2013, 9. Dostupné z: doi:10.2174/15734056113096660003
- [2] ZITOVÁ, Barbara a Jan FLUSSER. Image registration methods: a survey: a survey. *Image and Vision Computing*. 2003, 21(11), 977-1000. ISSN 0262-8856. Dostupné z: doi:https://doi.org/10.1016/S0262-8856(03)00137-9
- [3] DAMAS, S., Oscar CORDON a José SANTAMARÍA. Medical Image Registration Using Evolutionary Computation: An Experimental Survey: An Experimental Survey. *Computational Intelligence Magazine, IEEE*. 2011, 6, 26-42. Dostupné z: doi:10.1109/MCI.2011.942582
- [4] JAN, Jiří. *Medical Image Processing, Reconstruction and Restoration: Concepts and Methods*. Second. 6000 Broken Sound Parkway NW, Suite 300: Taylor and Francis Group, 2006. ISBN 0-8247-5849-8.
- [5] RUSINKIEWICZ, S. a M. LEVOY. Efficient variants of the ICP algorithm. In: *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*. 2001, s. 145-152. Dostupné z: doi:10.1109/IM.2001.924423
- [6] AIT-AOUDIA, Samy a Ramdane MAHIOU. *Medical Image Registration by Simulated Annealing and genetic algorithms*. 2007, 145 s. ISBN 0-7695-2901-1. Dostupné z: doi:10.1109/GMAI.2007.25
- [7] CHEN, Y., C. LIN a A. MIMORI. Multimodal Medical Image Registration Using Particle Swarm Optimization. In: *2008 Eighth International Conference on Intelligent Systems Design and Applications*. 2008, , s. 127-131. ISSN 2164-7151. Dostupné z: doi:10.1109/ISDA.2008.321
- [8] WACHOWIAK, M., R. SMOLIKOVA, Yufeng ZHENG, J. ZURADA a A. ELMAGHRABY. An approach to multimodal biomedical image registration utilizing particle swarm optimization. *IEEE Transactions on Evolutionary Computation*. 2004, 8(3), 289-301. ISSN 1941-0026. Dostupné z: doi:10.1109/TEVC.2004.826068
- [9] Hill DL, Batchelor PG, Holden M, Hawkes DJ. Medical image registration. *Phys Med Biol*. 2001 Mar;46(3):R1-45. doi: 10.1088/0031-9155/46/3/201. PMID: 11277237.

- [10] ROCHE, Alexis, Xavier PENNEC, Michael RUDOLPH, Dorothee AUER, Grégoire MALANDAIN, Sébastien OURSELIN, Ludwig M. AUER a Nicholas AYACHE. Generalized Correlation Ratio for Rigid Registration of 3D Ultrasound with MR Images. 2000//, s. -577, 567 s. Dostupné z: doi:10.1007/978-3-540-40899-4-58
- [11] MALIK, Sarfraz Ahmed, Rakesh Singh KUNWAR a M. Ehtashamoul HAQUE. Automatic image registration using evolutionary algorithm. Recent Research in Science and Technology [online]. 2011, 2011, 2011(4(1), 33-39 [cit. 2021-5-10]. ISSN 2076-5061. Dostupné z: <http://recent-science.com/>
- [12] CAGNONI, Stefano, Evelyne LUTTON a Gustavo OLAGUE. Genetic and evolutionary computation for image processing and analysis. 8. 2007/01/01. ISBN 1595931864.
- [13] GARZELLI, Andrea, Luca CAPOBIANCO a Filippo NENCINI. 9 - Fusion of multispectral and panchromatic images as an optimisation problem. *Oxford: Academic Press*, 2008, s. 223-250. ISBN 978-0-12-372529-5. Dostupné z: doi:<https://doi.org/10.1016/B978-0-12-372529-5.00005-6>
- [14] Kozumplík, Jiří. Evoluční algoritmy - přednáška 3. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2019.
- [15] POLYAK, Boris. Newton's method and its use in optimization. *European Journal of Operational Research*. 2007/09/16, 181, 1086-1096. Dostupné z: doi:10.1016/j.ejor.2005.06.076
- [16] LEPORÉ, Natasha, Yi-Yu CHOU, Oscar L LOPEZ, Howard J AIZENSTEIN, James T BECKER, Arthur W TOGA a Paul M THOMPSON. Fast 3D Fluid Registration of Brain Magnetic Resonance Images. *Proceedings of SPIE-the International Society for Optical Engineering*. 2008/03/, 6916, 69160Z. ISSN 0277-786X. Dostupné z: doi:10.1117/12.774338
- [17] SOTIRAS, Aristeidis, Christos DAVATZIKOS a Nikos PARAGIOS. Deformable medical image registration: a survey: a survey. *IEEE transactions on medical imaging*. 2013/07/, 32(7), 1153-1190. ISSN 1558-254X. Dostupné z: doi:10.1109/TMI.2013.2265603
- [18] CHE, Chengqian, Tejas Sudharshan MATHAI a John GALEOTTI. Ultrasound registration: A review: A review. *Methods*. 2017, 115, 128-143. ISSN 1046-2023. Dostupné z: doi:<https://doi.org/10.1016/j.ymeth.2016.12.006>

- [19] ZHANG, Jun, Guanchang JIN, Shaopeng MA a Libo MENG. Application of an improved subpixel registration algorithm on digital speckle correlation measurement. *Optics and Laser Technology*. 2003, 35(7), 533-542. ISSN 0030-3992. Dostupné z: doi:[https://doi.org/10.1016/S0030-3992\(03\)00069-0](https://doi.org/10.1016/S0030-3992(03)00069-0)
- [20] J. Woo, B.W. Hong, C.H. Hu, K.K. Shung, C.C.J. Kuo, P.J. Slomka, Non-rigid ultrasound image registration based on intensity and local phase information, *J. Signal Process. Syst.* 54 (2009) 33–43, dostupné z: <http://dx.doi.org/10.1007/s11265-008-0218-2>.
- [21] WANG, Z., G. SLABAUGH, G. UNAL a T. FANG. REGISTRATION OF ULTRASOUND IMAGES USING AN INFORMATION-THEORETIC FEATURE DETECTOR. *In: 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. 2007, s. 736-739. ISSN 1945-8452. Dostupné z: doi:10.1109/ISBI.2007.356957
- [22] MATINFAR, B. a L. ZAGROCHEV. Non-rigid Registration of 3D Ultrasound Images Using Model-Based Segmentation. *In: 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2014, s. 323-328. ISSN 2160-7516. Dostupné z: doi:10.1109/CVPRW.2014.132
- [23] Lu X, Zhang S, Yang W, Chen Y. SIFT and shape information incorporated into fluid model for non-rigid registration of ultrasound images. *Comput Methods Programs Biomed.* 2010 Nov;100(2):123-31. Dostupné z doi: 10.1016/j.cmpb.2010.03.005. Epub 2010 Apr 4. PMID: 20371129.
- [24] Heyde B, Alessandrini M, Hermans J, Barbosa D, Claus P, D’hooge J. Anatomical Image Registration Using Volume Conservation to Assess Cardiac Deformation From 3D Ultrasound Recordings. *IEEE Trans Med Imaging*. 2016 Feb;35(2):501-11. doi: 10.1109/TMI.2015.2479556. Epub 2015 Sep 17. PMID: 26394416.
- [25] Cen F., Jiang Y., Zhang Z., Tsui H.T., Lau T.K., Xie H. (2004) Robust Registration of 3-D Ultrasound Images Based on Gabor Filter and Mean-Shift Method. In: Sonka M., Kakadiaris I.A., Kybic J. (eds) *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis. MMBIA 2004, CVAMIA 2004. Lecture Notes in Computer Science*, vol 3117. Springer, Berlin, Heidelberg. Dostupné z: <https://doi.org/10.1007/978-3-540-27816-026>
- [26] Jansson T, Persson HW, Lindström K. Estimation of blood perfusion using ultrasound. *Proceedings of the Institution of Mechanical Engineers, Part*

- H: Journal of Engineering in Medicine. 1999;213(2):91-106. Dostupné z: doi:10.1243/0954411991534834
- [27] QUAIA, Emilio. Assessment of tissue perfusion by contrast-enhanced ultrasound. European Radiology. 2011, 21(3), 604-615. ISSN 1432-1084. Dostupné z: doi:10.1007/s00330-010-1965-6
- [28] LIU, Ji-Bin, Gervais WANSAICHEONG, Daniel A. MERTON, Flemming FORSBERG a Barry B. GOLDBERG. Contrast-enhanced Ultrasound Imaging: State of the Art: State of the Art. Journal of Medical Ultrasound. 2005, 13(3), 109-126. ISSN 0929-6441. Dostupné z: doi:https://doi.org/10.1016/S0929-6441(09)60100-4
- [29] WEISE, Thomas. Global Optimization Algorithm: Theory and Application: Theory and Application. 2009/01/01.
- [30] Kozumplík, Jiří. Evoluční algoritmy - přednáška 15. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2014.
- [31] HUSSEIN, Wasim A., Siti Norul Huda Sheikh ABDULLAH a Shahnorbanun SAHRAN. The Patch-Levy-Based Bees Algorithm Applied to Dynamic Optimization Problems. Discrete Dynamics in Nature and Society. Hindawi, 2017, 2017, 5678393. ISSN 1026-0226. Dostupné z: doi:10.1155/2017/5678393
- [32] TIACHAROEN, Somporn, S. POTIYA a Prakornchai PHONRATTANASAK. Solving various types of Economic Dispatch problem using bees algorithm. 2010/01/01.
- [33] Hsing-Chih Tsai, Integrating the artificial bee colony and bees algorithm to face constrained optimization problems, Information Sciences, Volume 258, 2014, Pages 80-93, ISSN 0020-0255, Dostupné z: https://doi.org/10.1016/j.ins.2013.09.015.
- [34] D R, Sarvamangala a Raghavendra KULKARNI. A Comparative Study of Bio-inspired Algorithms for Medical Image Registration. 2019/01/01, s. 27-44. ISBN 978-981-10-8973-2. Dostupné z: doi:10.1007/978-981-10-8974-9-2
- [35] MIRJALILI, Seyedali. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Computing and Applications. 2016, 27(4), 1053-1073. ISSN 1433-3058. Dostupné z: doi:10.1007/s00521-015-1920-1

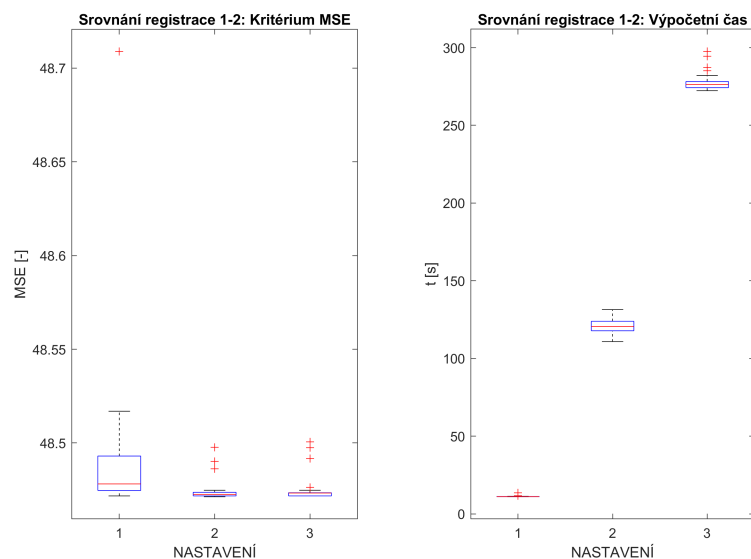
- [36] RUSSELL, Robert, Michael MAY, Kenneth SOLTESZ a John FITZPATRICK. Massive Swarm Migrations of Dragonflies (Odonata) in Eastern North America. The American Midland Naturalist. 2009/01/20, 140, 325-342. Dostupné z: doi:10.1674/0003-0031(1998)140[0325:MSMODO]2.0.CO;2
- [37] Craig W. Reynolds. 1987. Flocks, herds and schools: A distributed behavioral model. In Proceedings of the 14th annual conference on Computer graphics and interactive techniques (SIGGRAPH '87). Association for Computing Machinery, New York, NY, USA, 25-34. Dostupné z doi: <https://doi.org/10.1145/37401.37406>
- [38] RAHMAN, Chnoor a Tarik RASHID. Dragonfly Algorithm and its Applications in Applied Science -Survey. 2019/11/26.
- [39] MERAIHI, Yassine, Amar RAMDANE-CHERIF, Dalila ACHELI a Mohamed MAHSEUR. Dragonfly algorithm: a comprehensive review and applications: a comprehensive review and applications. Neural Computing and Applications. 2020/11/01, 32. Dostupné z: doi:10.1007/s00521-020-04866-y
- [40] ÖZBAKIR, Lale, Adil BAYKASOĞLU a Pınar TAPKAN. Bees algorithm for generalized assignment problem. Applied Mathematics and Computation. 2010, 215(11), 3782-3795. ISSN 0096-3003. Dostupné z: doi:<https://doi.org/10.1016/j.amc.2009.11.018>
- [41] ZIARATI, Koorush, Reza AKBARI a Vahid ZEIGHAMI. On the performance of bee algorithms for resource-constrained project scheduling problem. Applied Soft Computing. 2011, 11(4), 3720-3733. ISSN 1568-4946. Dostupné z: doi:<https://doi.org/10.1016/j.asoc.2011.02.002>
- [42] Seyedali Mirjalili (2021). Dragonfly Algorithm Toolbox (<https://www.mathworks.com/matlabcentral/fileexchange/51031-dragonfly-algorithm-toolbox>), MATLAB Central File Exchange. Retrieved May 9, 2021.
- [43] SINGER, S. a J. NELDER. Nelder-Mead algorithm. Scholarpedia. 2009, 4, 2928. Dostupné také z: *http : //www.scholarpedia.org/article/Nelder – Meadalgorithm*, doi::10.4249/scholarpedia.2928
- [44] Kozumplík, Jiří. Evoluční algoritmy - přednáška 5. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2019.
- [45] NELDER, J. A. a R. MEAD. A Simplex Method for Function Minimization. The Computer Journal. 1965, 7(4), 308-313. ISSN 0010-4620. Dostupné z: doi:10.1093/comjnl/7.4.308

Seznam příloh

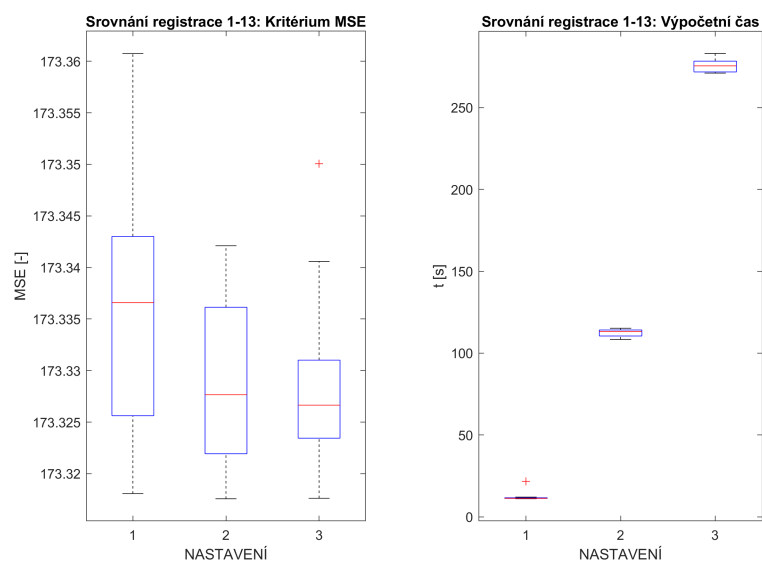
A	Vyhodnocení optimalizace na simulovaných datech	78
A.1	Algoritmus PSO	78
A.2	Algoritmus BA	80
A.3	Algoritmus DA	82
B	Obsah odevzdaných příloh	85

A Vyhodnocení optimalizace na simulovaných datech

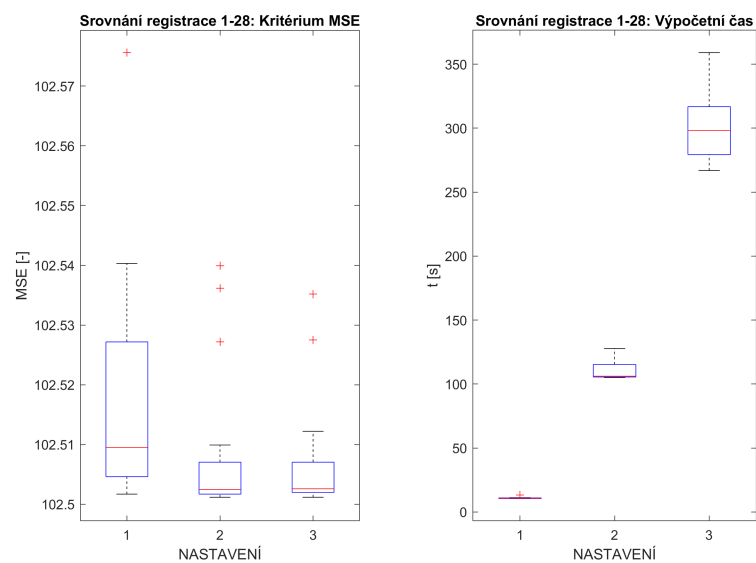
A.1 Algoritmus PSO



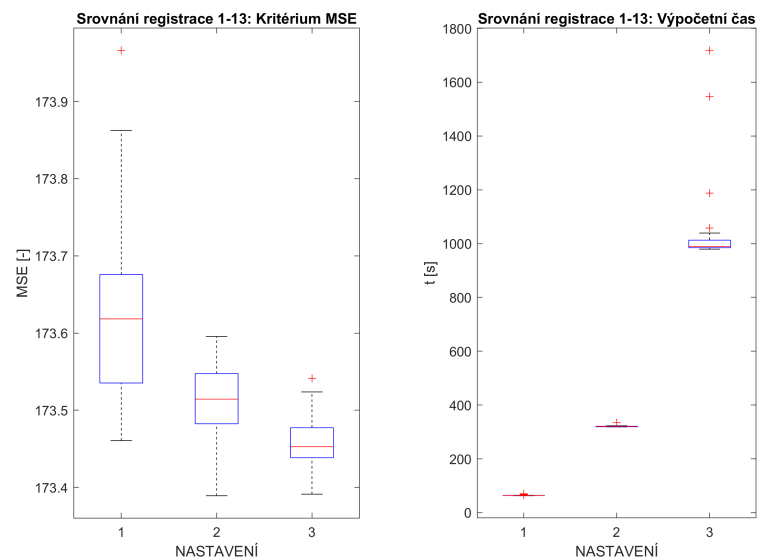
Obr. A.1: Registrace dvojice snímků 1-2, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost



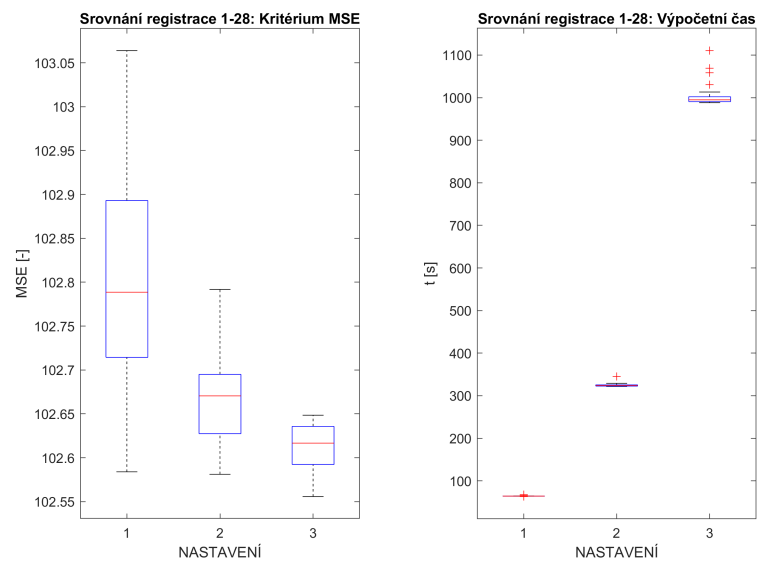
Obr. A.2: Registrace dvojice snímků 1-13, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost



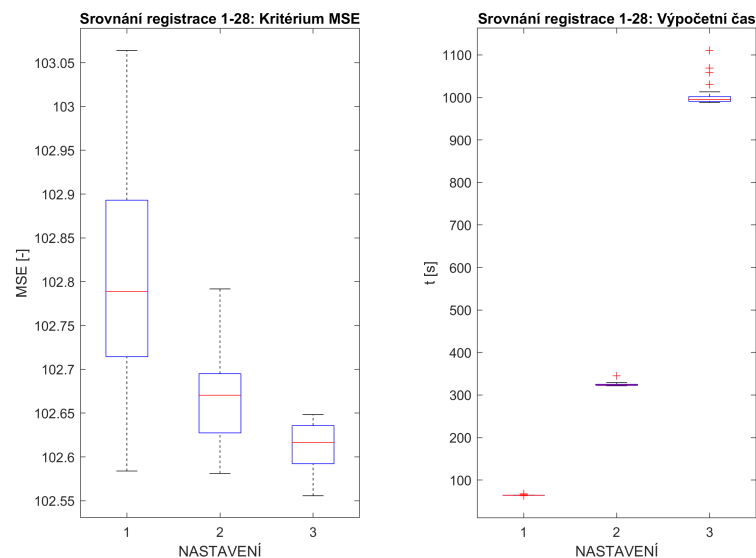
Obr. A.3: Registrace dvojice snímků 1-28, box plot vlevo: na ose x nastavení z tabulky 4.4, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.4, na ose y časová výpočetní náročnost



Obr. A.6: Registrace dvojice snímků 1-13, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost

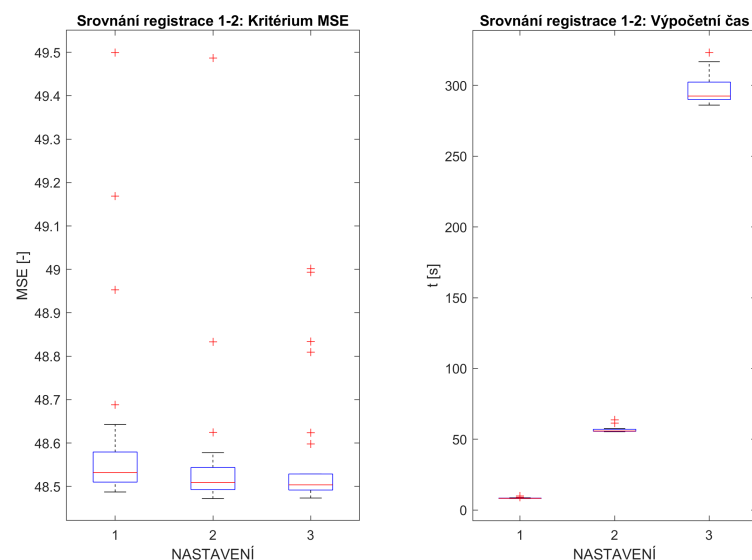


Obr. A.7: Registrace dvojice snímků 1-28, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost

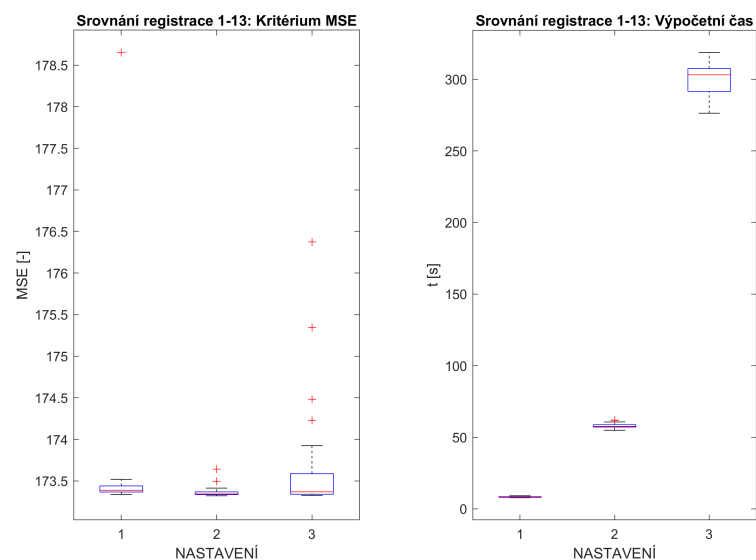


Obr. A.8: Registrace dvojice snímků 12-13, box plot vlevo: na ose x nastavení z 4.7, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.7, na ose y časová výpočetní náročnost

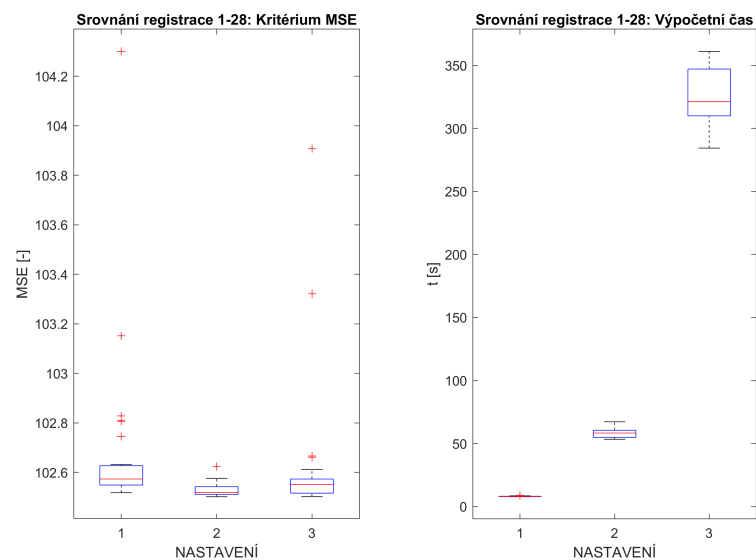
A.3 Algoritmus DA



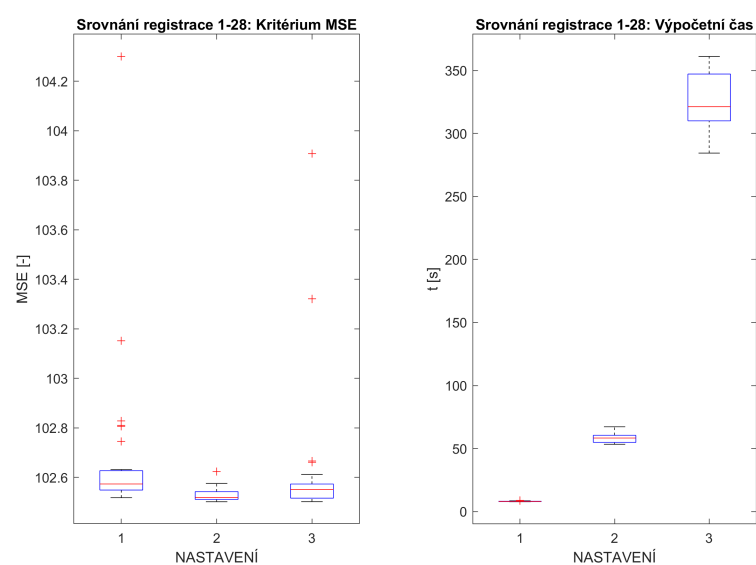
Obr. A.9: Registrace dvojice snímků 1-2, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost



Obr. A.10: Registrace dvojice snímků 1-13, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost



Obr. A.11: Registrace dvojice snímků 1-28, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost



Obr. A.12: Registrace dvojice snímků 12-13, box plot vlevo: na ose x nastavení z tabulky 4.10, na ose y hodnota kritériální funkce, box plot vpravo: na ose x nastavení z tabulky 4.10, na ose y časová výpočetní náročnost

B Obsah odevzdaných příloh

V přiloženém souboru naleznete implementované optimalizační algoritmy. Kódy pro tuto diplomovou práci byly testovány softwaru Matlab R2020a.

/..... Soubory DP Hnízdilová

- DATA
 - Puvodni_sekvence.mat
 - Simulovana_data.mat
 - Preklinicka_data.mat
 - Klinicka_data.mat
 - Klinicka_data2.mat
- optimalizace
 - BA_MSE.mat
 - PSO_MSE.mat
 - DA_.mat
 - Levy.mat
 - inicialization.mat
 - distance.mat
 - opt_MATLAB2.mat
 - opt_MATLAB3.mat
- registrace
 - registrace_BA.mat
 - registrace_DA.mat
 - registrace_PSO.mat
 - BA_MSE.mat
 - PSO_MSE.mat
 - DA_.mat
 - Levy.mat
 - inicialization.mat
 - distance.mat
- excel
 - TABULKA_1.xls
 - TABULKA_2.xls
- README.txt